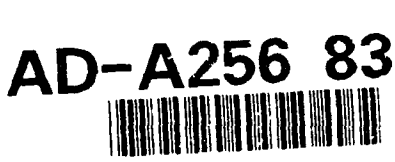


(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

UPGRADE AND EXTENSION OF THE
DATA ACQUISITION SYSTEM FOR
PROPULSION AND GAS DYNAMIC LABORATORIES

by

Richard A. Wendland

June, 1992

Thesis Advisor: Raymond P. Shreeve

Approved for public release; distribution is unlimited.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY Multiple Sources	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) AA	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)
UPGRADE AND EXTENSION OF DATA ACQUISITION SYSTEM FOR PROPULSION AND GAS DYNAMIC LABORATORIES

12. PERSONAL AUTHOR(S)
Wendland, Richard A.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM _____ TO _____	June 1992	207

16 SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government.

17 COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Data Acquisition System, HP-IB Compatible
High-speed Pressure Measurements

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The goal of the present work was to upgrade the data acquisition system (DAS) in the high-speed building of the Turbopropulsion Laboratory (TPL) and to develop a high-speed acquisition capability for pressure measurements for both TPL and the new Gas Dynamics Laboratory (GDL). Based on the use of the Hewlett Packard HP9000 Series 300 Computer as the system controller, a 96-channel high-speed pressure DAS was developed using Scanivalve ZOC-14 modules and a CALSYS2000 calibrator. The system allowed acquisition times for current wind-tunnel experiments to be revised from four minutes to eleven seconds. Also, new software was written to acquire data from existing rotary pneumatic Scanivalves and HP-IB compatible instrumentation so that all other existing acquisition capabilities were maintained in both laboratories.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Raymond P. Shreeve	22b TELEPHONE (Include Area Code) (408) 646-2593 22c OFFICE SYMBOL AA/SF

DD Form 1473, JUN 86 *Previous editions are obsolete* SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603 UNCLASSIFIED

Approved for public release; distribution is unlimited.

Upgrade and Extension of the Data Acquisition System for Propulsion and Gas Dynamic Laboratories

by

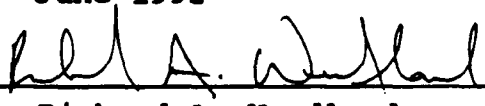
Richard A. Wendland
Lieutenant Commander, United States Navy
B.S.A.E., United States Naval Academy, 1980

Submitted in partial fulfillment
of the requirements for the degree of

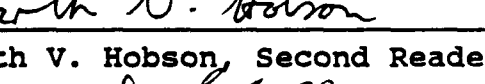
MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

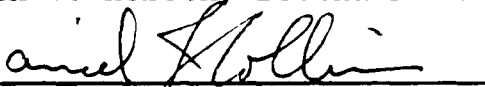
from the

NAVAL POSTGRADUATE SCHOOL
June 1992

Author: 
Richard A. Wendland

Approved by: 
Raymond P. Shreeve, Thesis Advisor


Garth V. Hobson, Second Reader


Dr. Daniel J. Collins, Chairman
Department of Aeronautics and Astronautics

ABSTRACT

The goal of the present work was to upgrade the data acquisition system (DAS) in the high-speed building of the Turbopropulsion Laboratory (TPL) and to develop a high-speed acquisition capability for pressure measurements for both the TPL and the new Gas Dynamics Laboratory (GDL). Based on the use of the Hewlett Packard HP9000 Series 300 Computer as the system controller, a 96-channel high-speed pressure DAS was developed using Scanivalve ZOC-14 modules and a CALSYS2000 calibrator. The system allowed acquisition times for current wind-tunnel experiments to be revised from four minutes to eleven seconds. Also, new software was written to acquire data from existing rotary pneumatic Scanivalves and HP-IB compatible instrumentation so that all other existing acquisition capabilities were maintained in both laboratories.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I extend my appreciation in acknowledging several Naval Postgraduate School Department of Aeronautics and Astronautics staff members whose efforts greatly contributed towards the development of this thesis.

I greatly appreciate the technical support provide by Rick Still in logistics, and the fabrication of unique components and devices to include the ZOC Enclosure assembly.

Recognition goes to Jack King as an electronic technician consultant, and the assembly of several electronic assemblies, notably the Auxiliary I/O Logic Interface device.

Last, a very special appreciation goes to Professor Ray Shreeve in providing the opportunity to pursue this thesis, and for his dedicated support in the thesis development.

I. INTRODUCTION

The Turbopropulsion Laboratory (TPL) consists of two large buildings, each with a central data acquisition system (DAS) serving multiple rotating and cascade test rigs. One building is for low-speed and one is for high-speed flow experiments. The Gas Dynamics Laboratory (GDL) consists of a single large building, closely adjacent to the other two, for which components for a central data acquisition system had been purchased, but not installed, at the inception of the present work. The GDL houses three blow-down wind tunnel facilities and a shock tube.

In planning the DAS for the GDL, commonality of the controller with the systems in the TPL was desirable in order to eliminate the need for students to learn different machines, to simplify the support task, and to interchange parts in the event of a failure. Since the TPL low-speed building was recently upgraded to use a Hewlett-Packard HP9000 Series 300 computer, two additional similar computers were purchased, one for the DAS new DAS and one to upgrade the DAS in the TPL high-speed laboratory. Thus the task in the present work was two-fold. First, software was required to be generated on the HP9000 which would enable all established DAS functions and experiments in the high-speed laboratory at the TPL to be maintained. Second, a high-speed scanning data

system was required to acquire pressure measurements in the short (minutes) duration blow-down wind tunnel tests in the GDL. Future upgrading in all laboratories would clearly depend on the experience gained in developing the high speed scanning system.

The pre-existing DAS's at the TPL incorporated Hewlett Packard HP-IB compatible scanners, digital voltmeters, system voltmeters, frequency counters and a locally developed HG-78K controller for pneumatic Scanivalves. Prior to the HP9000 acquisition, in earlier updates, the HP9830, HP9845 and HP1000 computers had been used as system controllers. Because of almost unlimited run-times of the rigs at the TPL, and the need largely to record many channels of "steady-state" measurements, the data-recording times were not a critical issue. [For "real-time" pressure measurements, a 16-channel 100 Khz capability was provided using a (non HP-IB) DMA input to the HP1000 from a now-obsolete HP5610A A/D converter].

Data acquisition was accomplished in the "steady-state" system, by computers executing individual data measurement instructions under program control. The maximum data collection rate was determined by the speed that the computers could execute individual instructions within the program between consecutive data measurement steps. P r e s s u r e measurements were made using the Scanivalve rotary pneumatic pressure sensing unit operated by the NPS HG-78K Controller. The Scanivalve unit sequentially stepped through its block of

ports collecting single pressure data once every 1.5 seconds. The HG-78K stepped the Scanivalve to the specified measurement port on commands executed by the computer. In summary, the pressure data acquisition rate was limited by the speed of the peripheral devices and by the computer's speed in executing program instructions.

The first task in the present work, to regenerate the software for the TPL "steady-state" DAS in HP BASIC 5.13 on the HP9000, served to become familiar with the computer and its programming, but no detailed report is included herein. The main task was to develop a high-speed scanning system for GDL, and this is documented in detail.

The hardware for the new system included the HP9000 Desk Top Computer System, HP6944A Multiprogrammer, Scanivalve ZOC-14 Electronic Scanning Pressure Module and CALSYS2000 Calibrator. The HP9000 serves as the computer-controller using the BASIC programming language. Data and program storage for the HP9000 is handled by a HP9153C Disc Drive incorporating a 40 mega-byte hard drive and 1.44 mega-byte 3.25" floppy drive. (The HP9153C replaced the older HP7906 Disc Drive and HP7970E Tape Drive of the HP1000 system).

The HP6944A combines several data acquisition devices (I/O Cards) into one unit. The significant feature of the HP6944A is its ability to perform specific data acquisition functions, and to control the data measurement steps without intervention from the host computer (HP9000). The HP6944A removes the

individual data measurement instructions of the program from the acquisition process, resulting in data rates which are limited only by the speed of the individual I/O card and the non-Hewlett Packard device.

In comparison with the TPL "steady-state" system, the ZOC-14 module replaces the low speed rotary Scanivalve and instead provides high speed electronic scanning of the pressure ports. The CALSYS2000 provides ZOC control and supplies calibration pressures to it, making the ZOC and CALSYS2000 a "packaged" pressure measurement system.

The software which was generated to integrate the HP9000, HP6944A and Scanivalve ZOC-14/CALSYS2000 system was designated as program SCAN_ZOC_05. The complete package of hardware and software is referred to *hereafter* as the ZOC-14 Data Acquisition System, or ZOC-14 DAS.

In the present document, Chapter Two describes and discusses the development of the ZOC-14 DAS. The hardware, individual component operation, integration of the components into a complete system, and the application programs to operate the DAS are discussed in detail. It is hoped that this chapter will serve as a manual for the system, and as a guide to those responsible for future extensions.

Chapter Three discusses the upgraded capabilities of TPL and GDL DAS's, potential extensions and outstanding issues. (The ZOC-14 DAS is fully operational, but requires resolution of these issues to optimize the system's performance).

Finally, conclusions are drawn and a particular recommendation is made in chapter Four.

Additional details are given in the Appendices. Appendix A contains the ZOC-14 DAS application program flow chart, the application program SCAN_ZOC_05, and associated utility programs. The utility programs are used to read and display the results of the data compiled by the SCAN_ZOC_05 program.

Appendix B contains four earlier versions in arriving at the SCAN_ZOC_05 program. The program evolved in steps, as knowledge was gained on the hardware operation and integration, and software programming. Each program is annotated with pertinent information to describe the program's functional routines, and are reported as and aid to future programming.

Appendix C contains a brief guide to managing data files provided by the ZOC-14 DAS collection process. A program is listed, and instructions provided, to view and delete selected data files from the hard drive. A sample print-out of the hard drive contents is provided, showing the data file listing and unique structure for the ZOC-14 DAS data files.

Appendix D provides a listing of all major programs developed for use on the HP9000 for the TPL. A compilation of all programs, listed in groups of "directories" contained on the HP9153C hard drive, is provided. The major program listings are divided into the categories of Turbocharger Performance, AE4431 Turbomachinery Design, and several sample

routines to operate devices, perform calculations, and manipulate files.

II. ZOC-14 DATA ACQUISITION SYSTEM

A. General Overview

1. Hardware

The ZOC-14 Data Acquisition System (DAS) hardware includes the Scanivalve ZOC-14 Electronic Pressure Scanning Module (Figure 1), the Scanivalve CALSYS2000 Calibration System (Figure 2), the Hewlett Packard HP6944A Multiprogrammer (Figure 3), and the Hewlett Packard HP9000 Desk Top Computer System and peripherals (Figure 4).

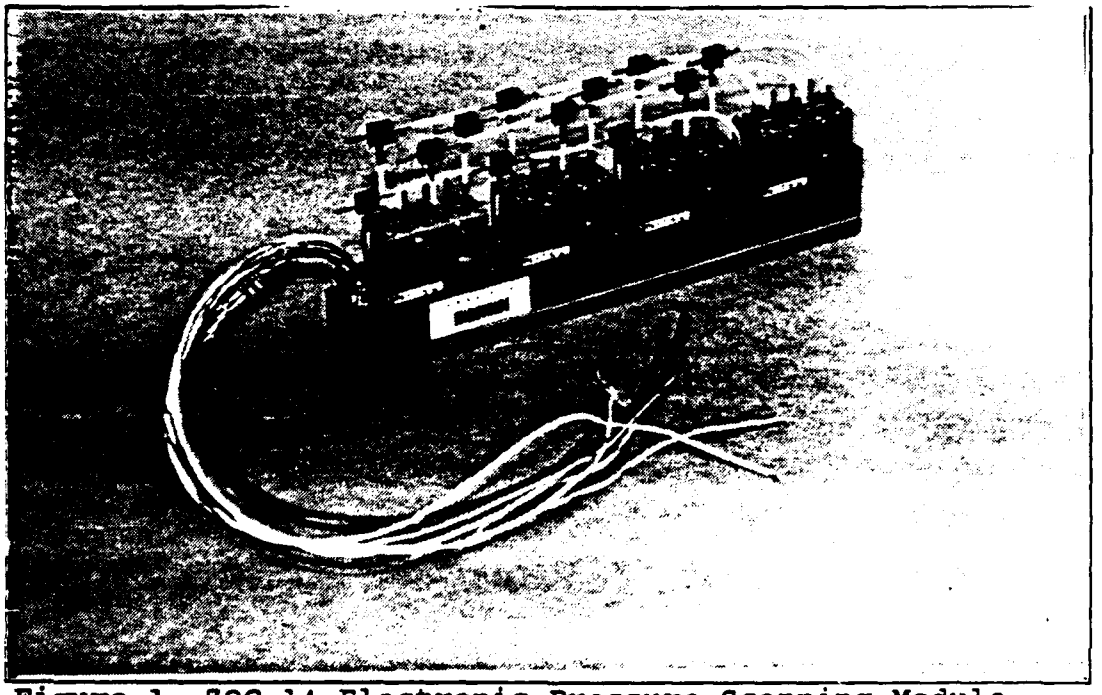


Figure 1 ZOC-14 Electronic Pressure Scanning Module

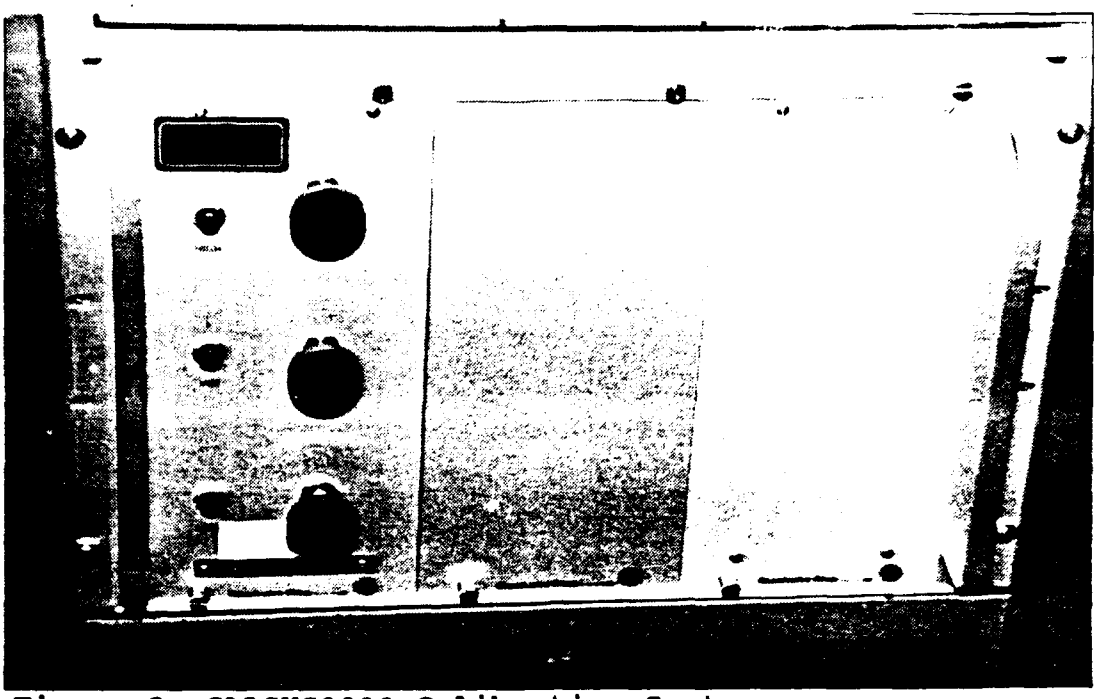


Figure 2 CALSYS2000 Calibration System

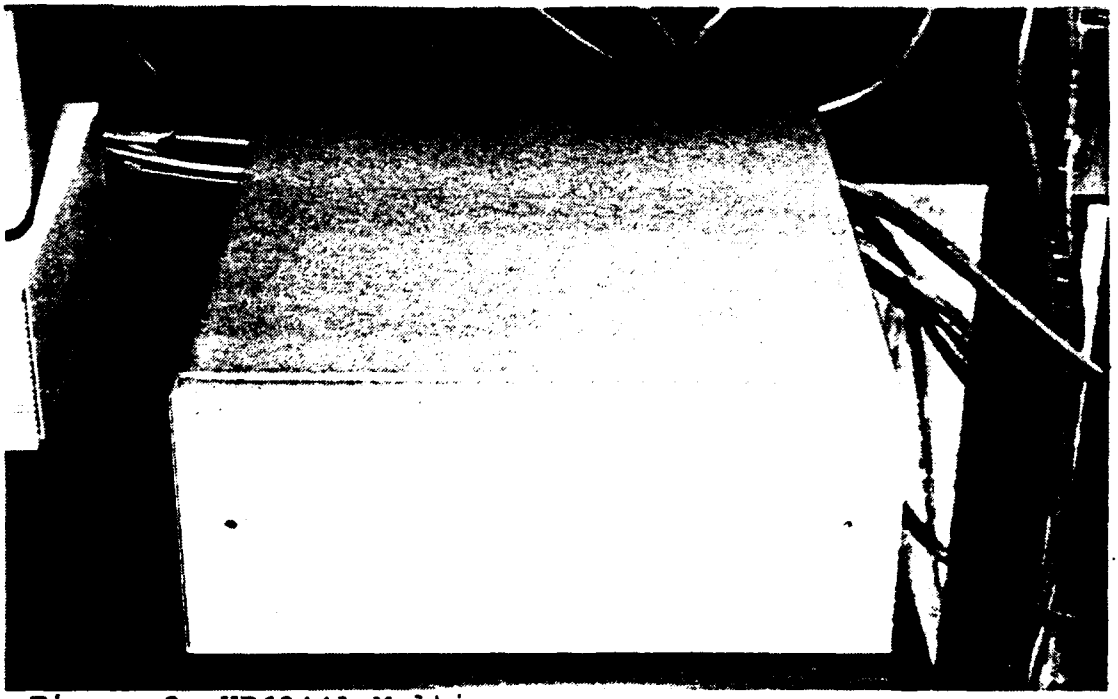


Figure 3 HP6944A Multiprogrammer

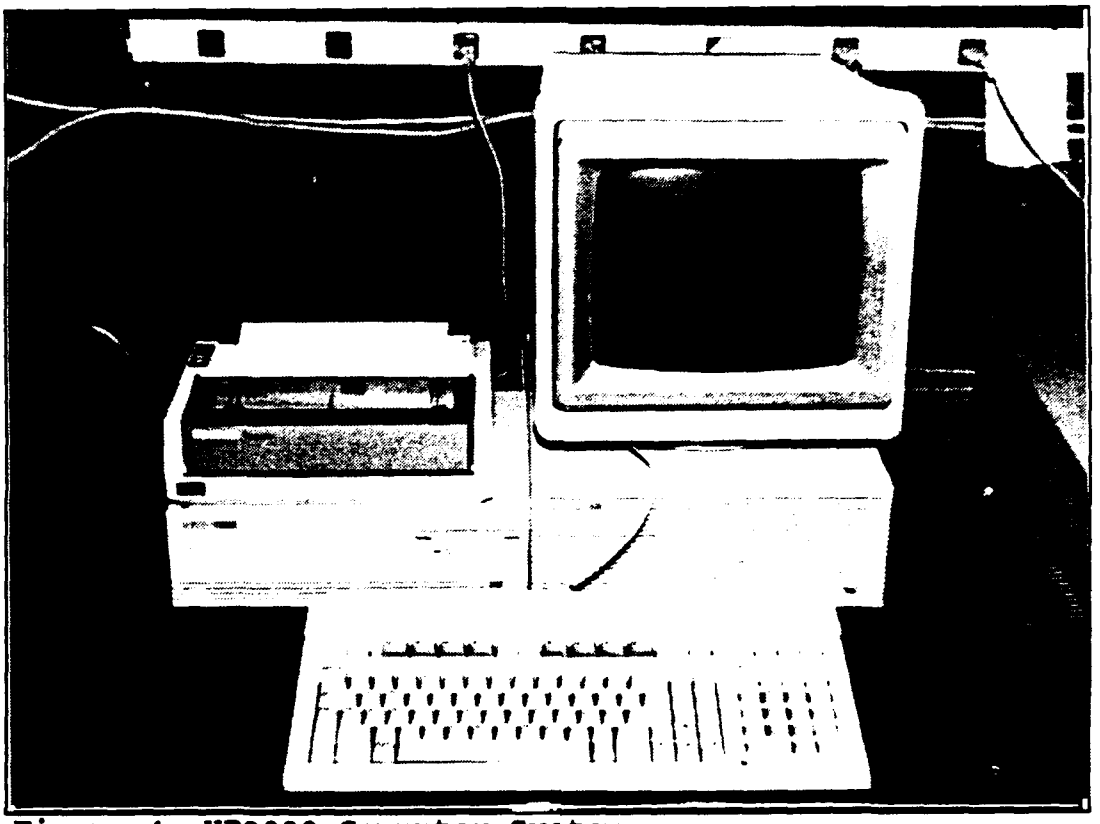


Figure 4 HP9000 Computer System

2. Software and Literature

The HP9000 is provided with BASIC 5.13 software. On-line compilation occurs when a program is executed by the RUN command. This user-friendly feature allows expeditious changes to be made in a program without the separate time consuming re-compilation process required by other high level programming languages. The result is the ability to RUN a program, change the program, and RUN the program again with minimal effort and time, making BASIC an ideal tool for an engineer.

The BASIC software system and documentation includes the necessary firmware provided on seven 3.25" floppy disks and twelve manuals describing the loading, utilization, and maintenance of BASIC (Figure 5)[Ref. 1 through Ref. 12].

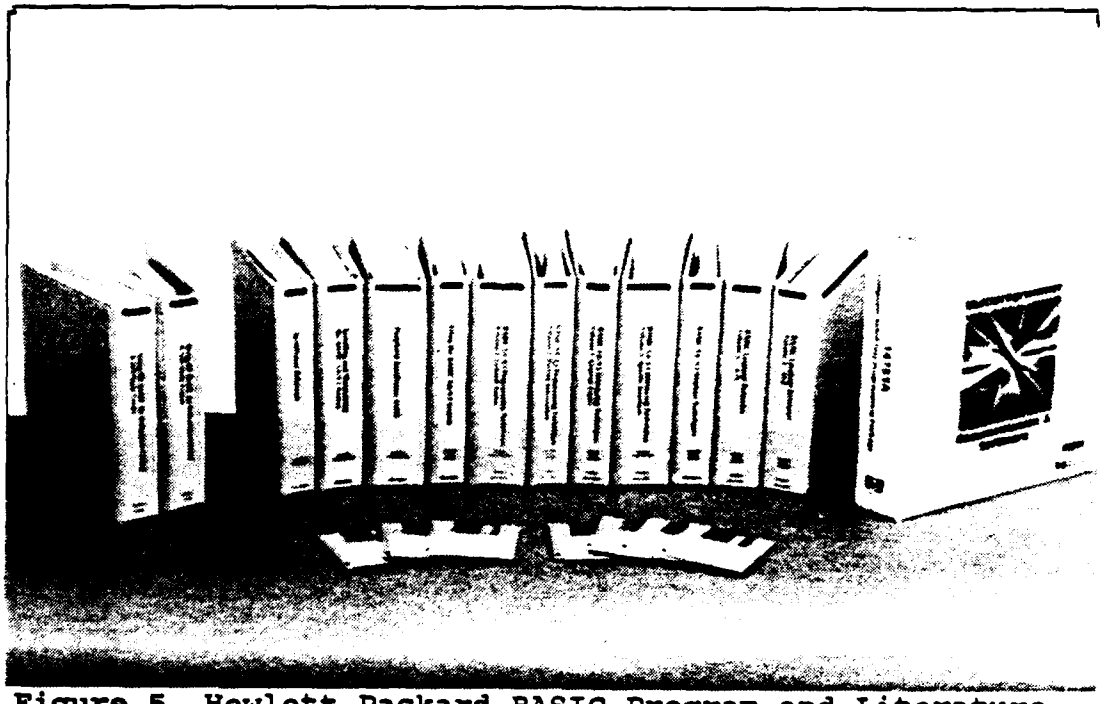


Figure 5 Hewlett Packard BASIC Program and Literature

The two volume Hewlett Packard Educational Package (Figure 5)[Ref. 13 through Ref. 14] is vital to quickly learn BASIC and its implementation with the HP9000 and the line of Hewlett Packard peripheral devices. This package condenses all the pertinent information contained in the BASIC program manuals. The novice programmer will find that the Educational Package provides the necessary fundamentals in how to use BASIC. The advanced programmer will be regularly referencing the BASIC program manuals for his programming requirements.

The HP14753A CAT Program Package (Figure 5)[Ref. 15] comprises the firmware and documentation required to set-up and operate the HP6944A Multiprogrammer. Details of this package will be discussed later in this chapter.

3. TPL ZOC-14 DAS System

The TPL ZOC-14 DAS System is an integration of all the aforementioned hardware, firmware, and newly developed software into an application package. The capability of the system is the ability to collect pressure data on multiple channels at high sampling rates, reduce the raw data, and store the reduced data using a user friendly, menu-driven operating program on the HP9000 computer. A schematic of the ZOC-14 DAS System is illustrated in Figure 6. The HP9000 computer is the central controlling device for the system. Software on the HP9000 controls data collection from the CALSYS2000 and HP6944A, data reduction, and data storage to the computer's hard drive and floppy drive. The HP6944A interfaces with the ZOC-14 modules providing electronic port selection, receiving analog voltages and converting to digital data. The CALSYS2000 is controlled by the HP9000, provides calibration pressure to the ZOCs, pneumatically sets the ZOC's operating modes, and provides digital data conversion of calibration pressure to the HP9000. The ZOC-14 module converts pneumatic pressures to analog voltage signals which are collected in the HP6944A.

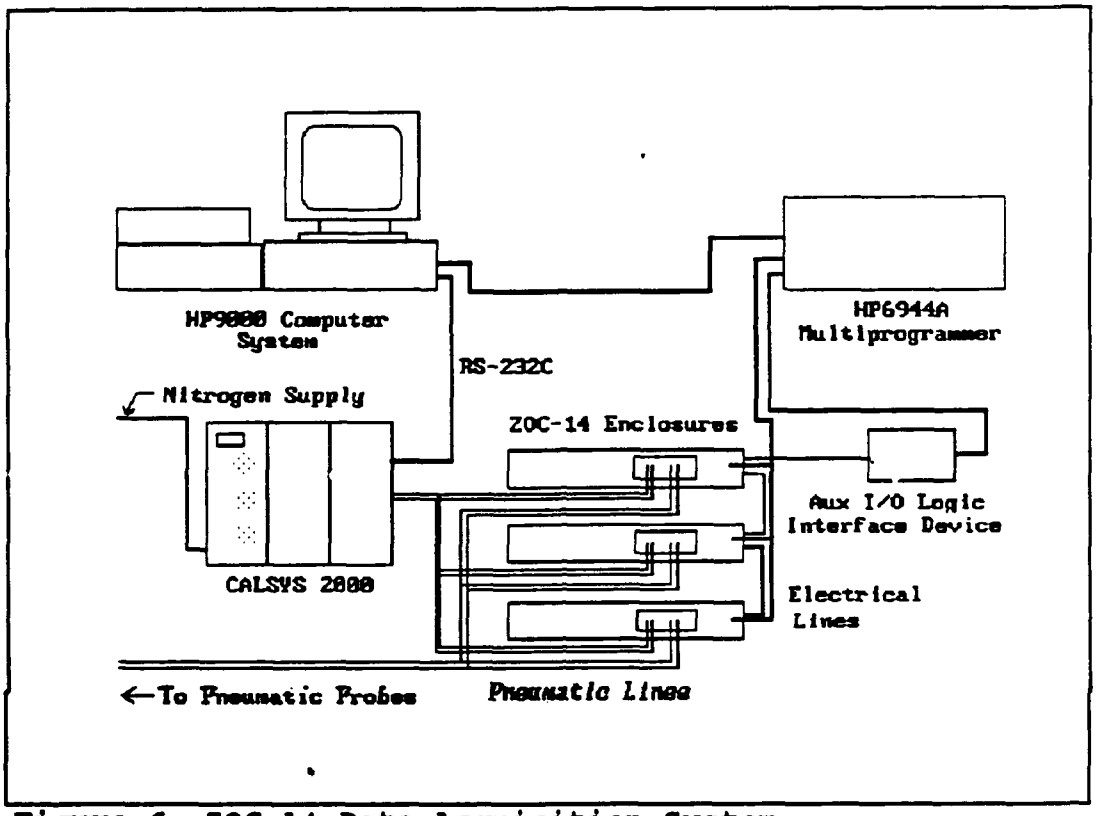


Figure 6 ZOC-14 Data Acquisition System

B. Hardware Description

1. ZOC-14 Electronic Pressure Scanning Module

a. Principle of Operation

The ZOC-14 is a 32 port electronically switched pressure sensing device. Internally, pressures are converted through semi-conductor strain gages to an analog voltage output signal for each selected port (Figure 7). The electronic switching feature allows the individual strain gages to be selected at random, and the output voltage signal to be read.

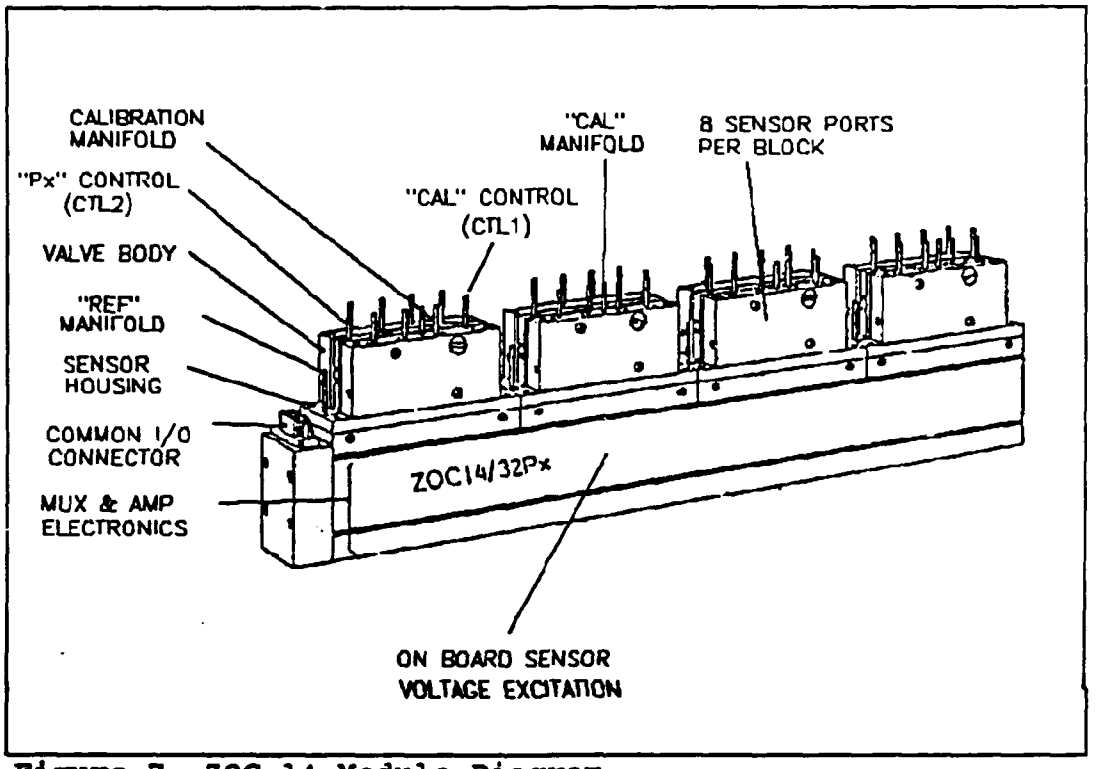


Figure 7 ZOC-14 Module Diagram

Each ZOC contains a series of valves that are pneumatically switched to set an operating mode. Shown in Figure 8 are the ports P1 through Px which are connected by pneumatic pressure tubing to the apparatus or probe wnere pressure is to be measured, such as the model in a wind tunnel. The Calibration Manifold and Sensor Reference Pressure Manifold are connected to the CALSYS2000 to receive a reference calibration pressure to be measured by the ZOC for calibration purposes. CAL Control (CTL1) and Px Control (CTL2) are pneumatic control lines that position the valves to set the required operating mode in the ZOC. CAL and Px control pressures are provided by the CALSYS2000.

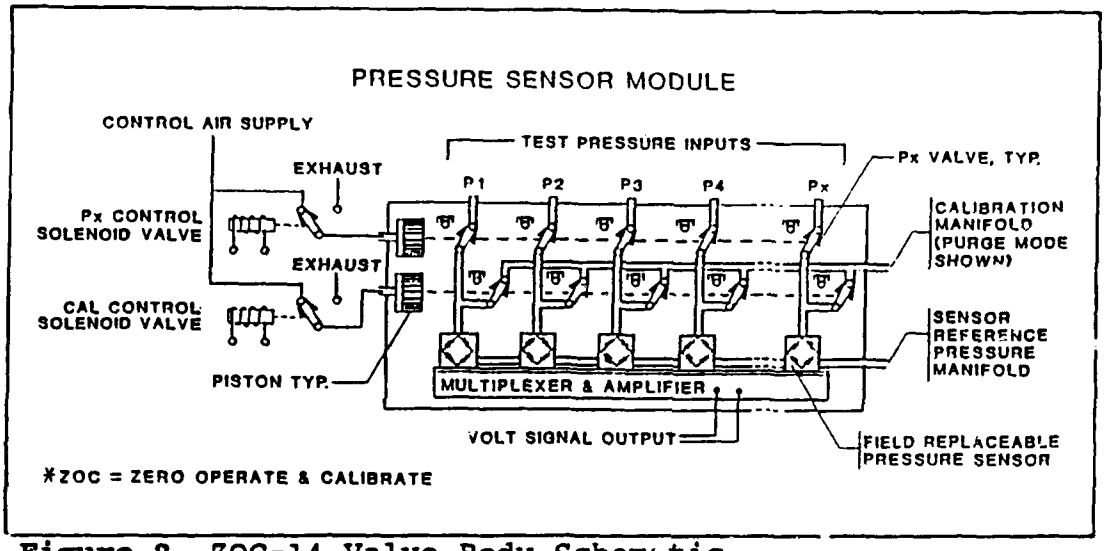


Figure 8 ZOC-14 Valve Body Schematic

The ZOC operates on the principle "ZERO OPERATE AND CALIBRATE"; hence the name "ZOC". This principle involves a two phase sequence. The first phase sets the ZOC into an "OPERATE" mode and experimental pressure data at pressure ports P1 through Px are converted and recorded. The second phase sets the ZOC into the "CALIBRATE" mode. The CALSYS2000 provides a calibration pressure through the Calibration Manifold and Sensor Reference Pressure Manifold to each strain gage. Calibration pressure data are then collected from the ZOC for each port by the HP6944A. The HP9000 collects the pressure data from the HP6944A and digital converted calibration pressure data from the CALSYS2000. These two sets of data are plotted against each other to get a calibration curve for each ZOC strain gage.

Figure 9 illustrates the ZOC's four modes of operation. In the OPERATE mode, P1 pressure is routed to the

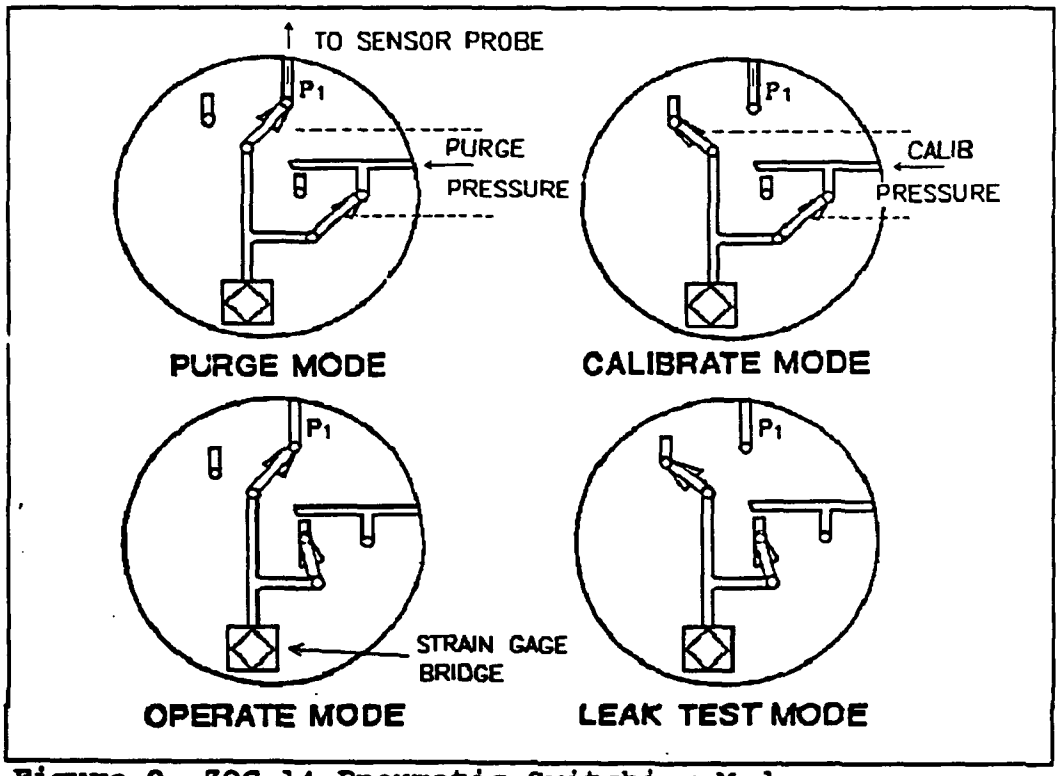


Figure 9 ZOC-14 Pneumatic Switching Modes

strain gage, and the Calibration Manifold is isolated. In the CALIBRATE mode, calibration pressure is routed to the strain gage, and the P1 pressure is isolated. The PURGE and LEAK TEST modes are used for clearing pressure lines and for diagnostic checks, respectively.

The Multiplexer and Amplifier section (Figure 8) facilitates the ZOC's high speed scanning capability. The multiplexer is driven by a five bit address input signal (A0 through A4) from the HP6944A, selecting a specified port from 1 to 32 in binary code. The ZOC's electronic module is illustrated schematically in Figure 10. The analog output signal processing is accomplished within the electronic

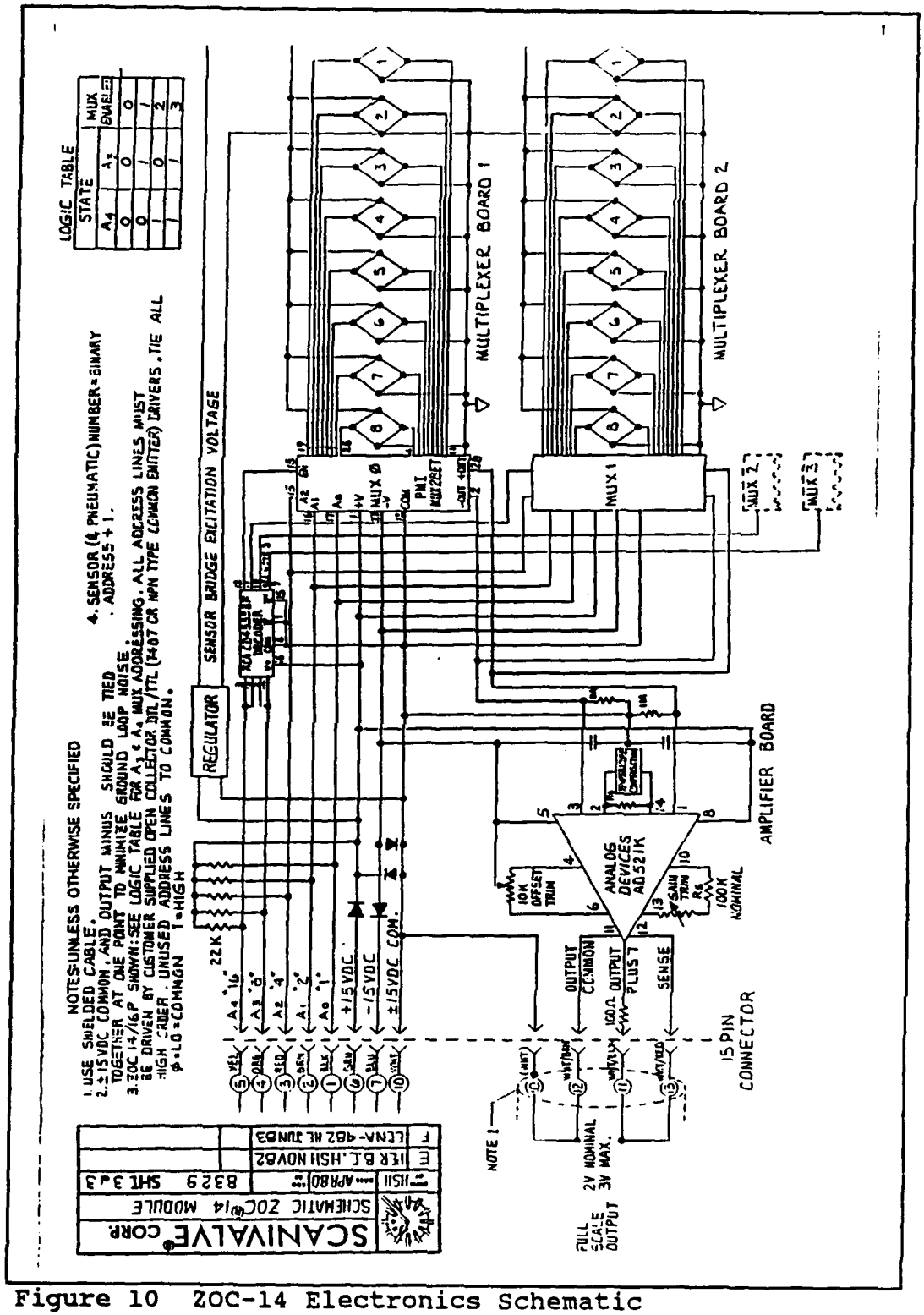


Figure 10 ZOC-14 Electronics Schematic

section of the ZOC.

Reference 16 provides all pertinent specifications, descriptions of the design and operation, and is the source document for the above technical information on the ZOC-14 Module.

b. ZOC-14 Enclosure

The ZOC-14 Enclosure was designed and built at NPS to provide a mobile and versatile module which was convenient for electronic and pneumatic connections, and provided protection in the laboratory environment. Components in the enclosure include one ZOC-14 Module (Figure 1), a 115 VAC power supply, a BNC connection for the output signal, Cannon plugs for the ZOC address input, and a pneumatic connection plate for measurement, control, and calibration pressure lines (Figure 11).

Each Px port on the ZOC-14 module has a corresponding port connection on the pneumatic connection plate. The connection plate port numbers 1 through 32 correspond to the ZOC's ports as selected by the binary address code. The Px Control, CAL Control, CAL, and REF ports on each valve block (Figure 7) are each connected to a common line as illustrated in Figure 12. The Px Control line is routed to port number 33, CAL Control to port number 34, CAL to port number 35, and REF to port number 36. Ports 37 through 48 are not used.

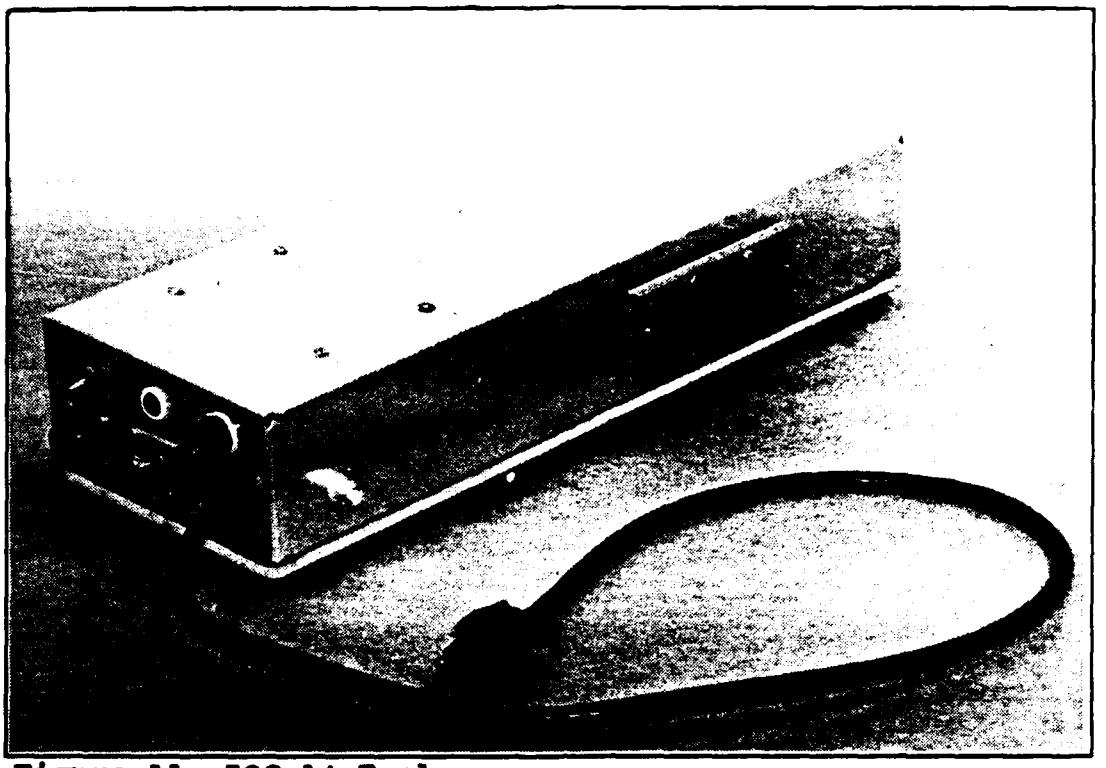


Figure 11 ZOC-14 Enclosure

The enclosure's electrical wiring is illustrated schematically in Figure 13. The two-five pin Cannon plugs are wired in parallel to jumper ZOC address line connection points between adjacent enclosures. This feature allows for one address line from the HP6944A to be connected to the first enclosure Cannon plug. The second enclosure receives its ZOC address signal from the second Cannon plug on the first enclosure by a jumper line. The third and following enclosures receive their ZOC address signals in the same sequence of connections.

A significant capability of the ZOC address control feature, in connection with the HP6944A, is the

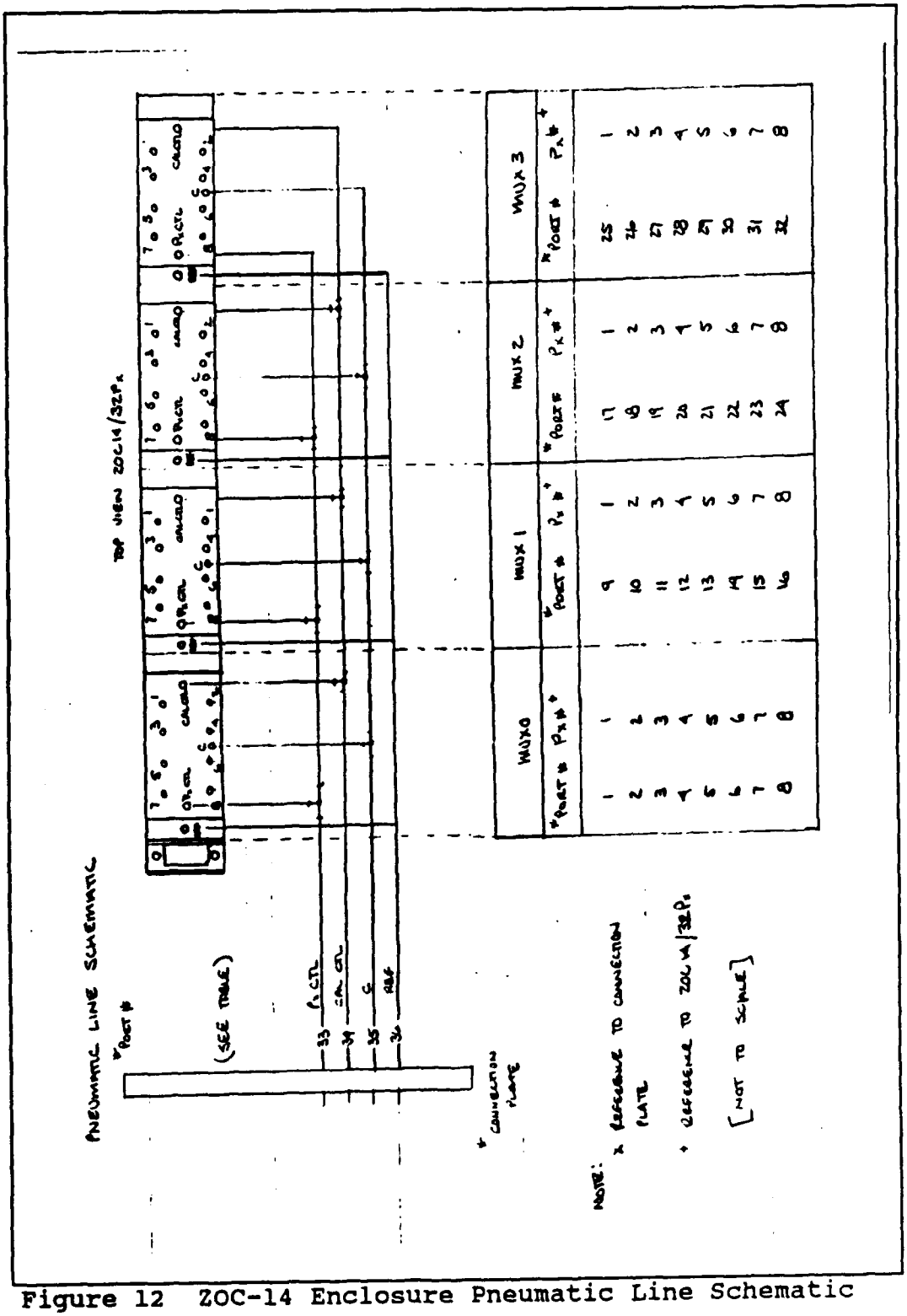


Figure 12 ZOC-14 Enclosure Pneumatic Line Schematic

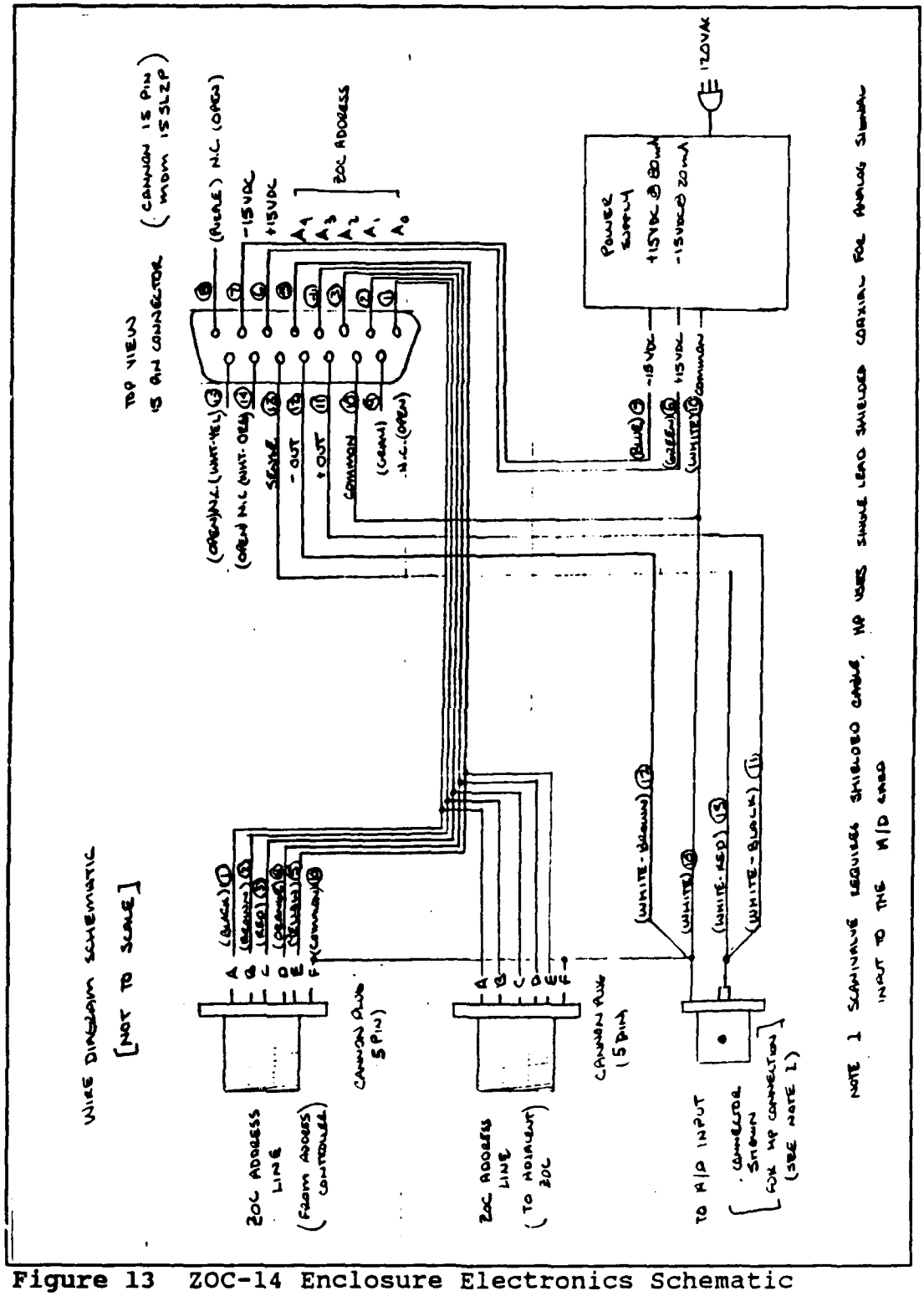


Figure 13 ZOC-14 Enclosure Electronics Schematic

ability to simultaneously collect pressure data from corresponding ports on a multiple ZOC configuration. This capability could play a significant role in determining the pressure behavior at two or more different points at the same instance in time. The number of instantaneous data points is determined by the number of ZOCs connected into the ZOC-14 DAS System.

2. CALSYS2000 Calibration Module

a. Description

The CALSYS2000 interfaces the ZOC module with the HP9000 computer by setting the ZOC mode selection as commanded by the HP9000, providing calibration information to the HP9000, and sending a reference calibration pressure to the ZOC for calibration purposes. Figure 14[1] shows a front view of the CALSYS2000 module.

The CALSYS2000 is comprised of a Calibration Module (CALMOD 2000) and a Power and Solenoid Control (PSC 2000) module. Figure 15 illustrates the external pneumatic and electrical line connections between the two modules, the nitrogen supply source and ZOC pneumatic line connections as seen in a rear view of the CALSYS2000 as it is installed at the TPL.

---

[1] Figure 14 is found in Ref. 17, page 10, as Figure 1.

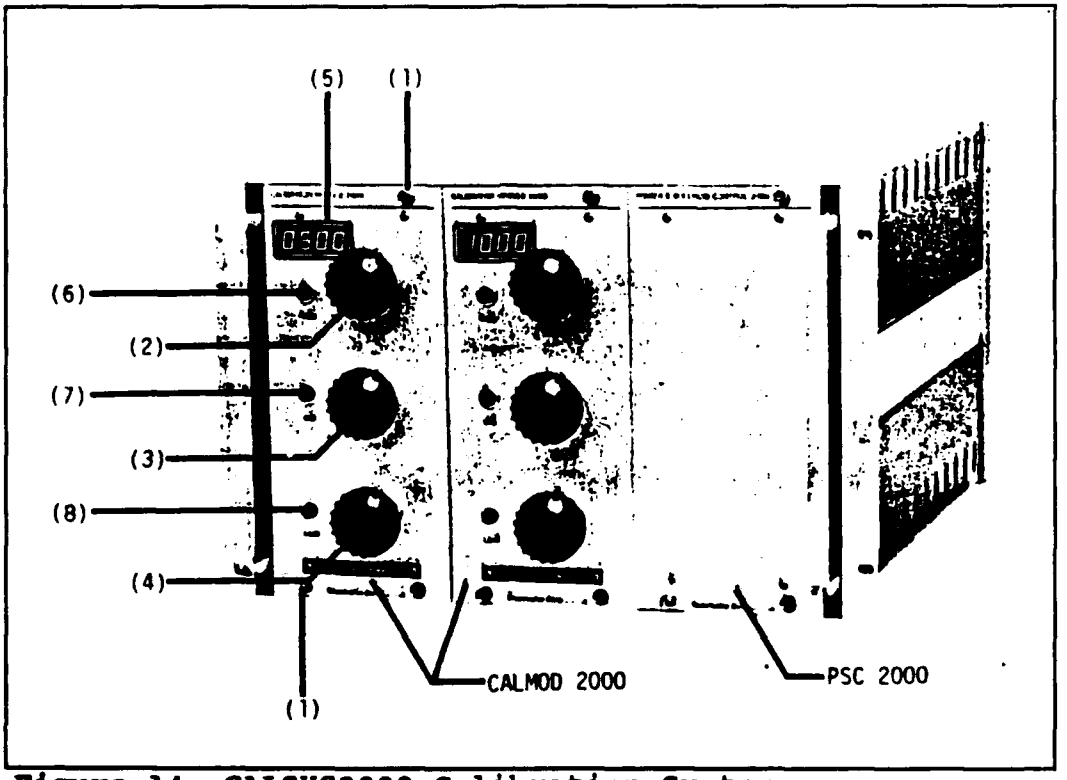


Figure 14 CALSYS2000 Calibration System

b. Calibrator Module (CALMOD 2000)

The CALMOD 2000 provides the electronic interface with the HP9000 through the serial RS-232C data link with a built-in processor. The processor responds to various commands from the HP9000 and issues control signals to actuate an array of solenoids within the CALMOD and PSC. The selective setting of the solenoids routes reduced nitrogen gas pressure to set the ZOC modes, and provide a regulated calibration reference pressure for ZOC transducer calibration. An internal Pressure Standard module samples the calibration pressure and provides a proportional analog voltage value to the processor. The processor's internal A/D and mathematical

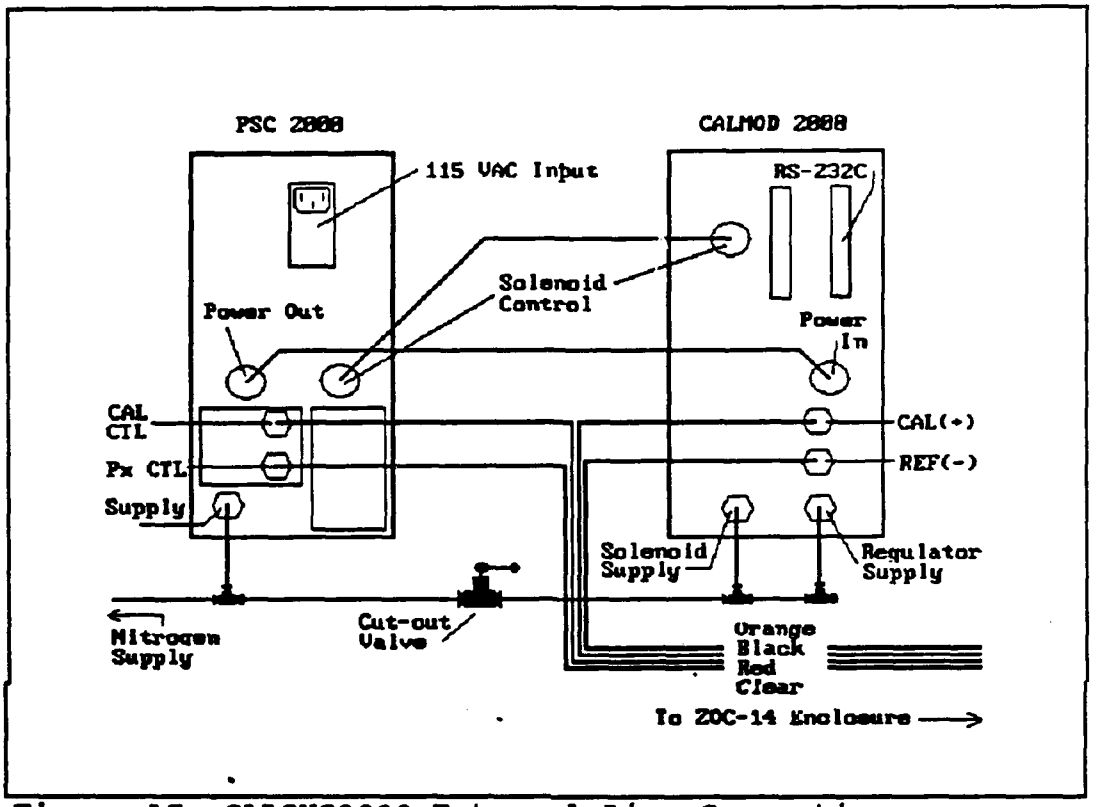


Figure 15 CALSYS2000 External Line Connections

processors return a digital measure of the pressure in either psia or inches of mercury. The CALMOD is currently set for inches of mercury. The processor provides this digital pressure value at the front window on the CALMOD (item 5, Figure 14), and also returns the value to the HP9000 following a "read pressure" command.

The CALMOD has three manually adjustable pressure regulators (items 6,7,8 in Figure 14) and associated pressure display buttons (items 2,3,4 in Figure 14). These regulators provide three different reference calibration pressures that the ZOC uses for calibration. During the calibration mode, each one of the three calibration pressures are sequentially

sent to the ZOC and the Pressure Standard. To calibrate the ZOC for positive pressures, the calibration pressure is provided through the Calibration Manifold to the strain gage transducers. To calibrate the ZOC for negative pressures, the calibration pressure is provided through the Sensor Reference Pressure Manifold to the backside of the strain gage transducer. Pressure data from the CALSYS2000 and ZOC are collected by the HP9000 and reduced to obtain calibration curves for each ZOC transducer.

Figure 16[2] shows the rear panel of the CALMOD 2000 with all the installed connection ports. Figure 15 illustrates the ports currently used at TPL. The CALMOD receives a common low-pressure supply of nitrogen gas to the Solenoid Supply and Regulator Supply connections. Regulated calibration pressure is supplied to the ZOC Calibration Manifold through the CAL(+) connection, and the Sensor Reference Pressure Manifold through the REF(-) connection.

Figure 17[3] shows the pneumatic lines between the CALSYS2000 and ZOC units. The TPL currently has two 15 psid and one 50 psid ZOCs. These ranges require a nominal 90 psi Regulated (Instrument) Air Supply to properly operate[4]. The solenoid supply is 90 psi for both the CALMOD and PSC.

---

[2] Figure 16 is found in Ref. 17, page 12, as Figure 2.

[3] Figure 17 is found in Ref. 17, page 21, as Figure 5.

[4] Refer to Ref. 17, dwg 16202, sht 1 of 9, in back of ref. 17.

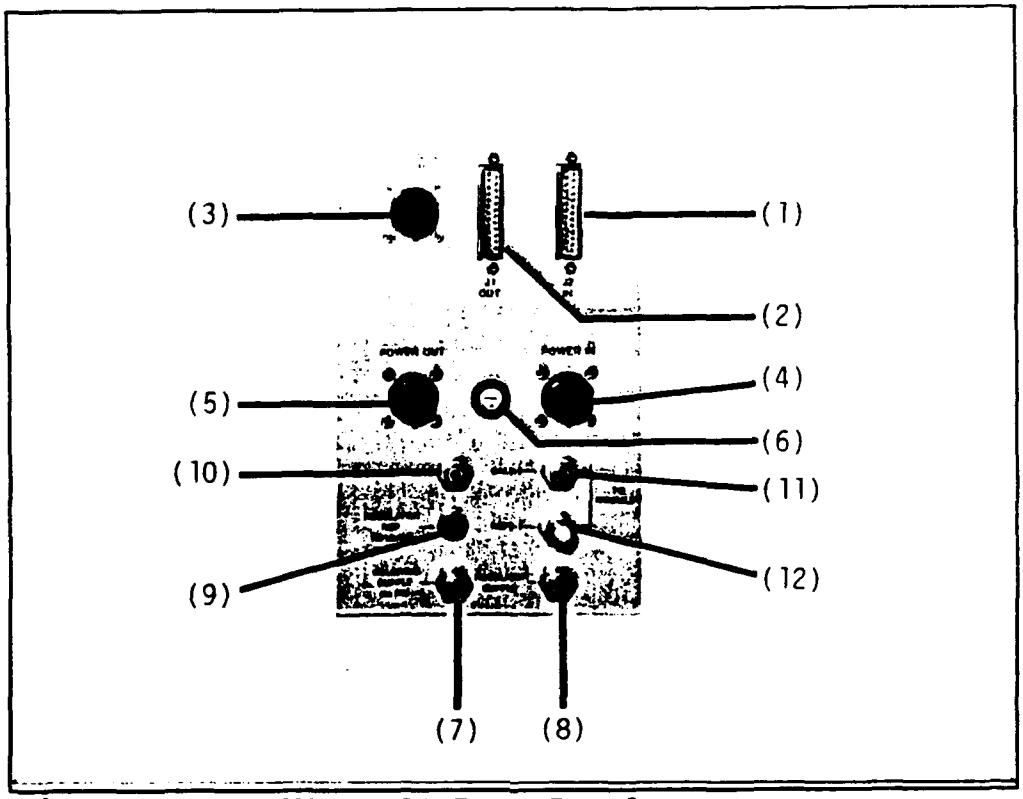


Figure 16 CALMOD 2000 Rear Panel

c. Power and Solenoid Control Module (PSC 2000)

The PSC provides the CALSYS2000 electrical power and pneumatically controls the ZOCs. Figure 18 shows the rear panel of the PSC with all external connections. Electrical power is supplied to the CALMOD through an electrical jumper between the Power Out (item 3 in Figure 18) and Power In terminals shown in Figure 15. The PSC contains the solenoids which route control air, CAL CTL and Px CTL (item 5 in Figure 18) to the ZOC for mode selection. These two solenoids are controlled by the CALMOD through an electrical cable connecting the Solenoid Control connections (item 3 in

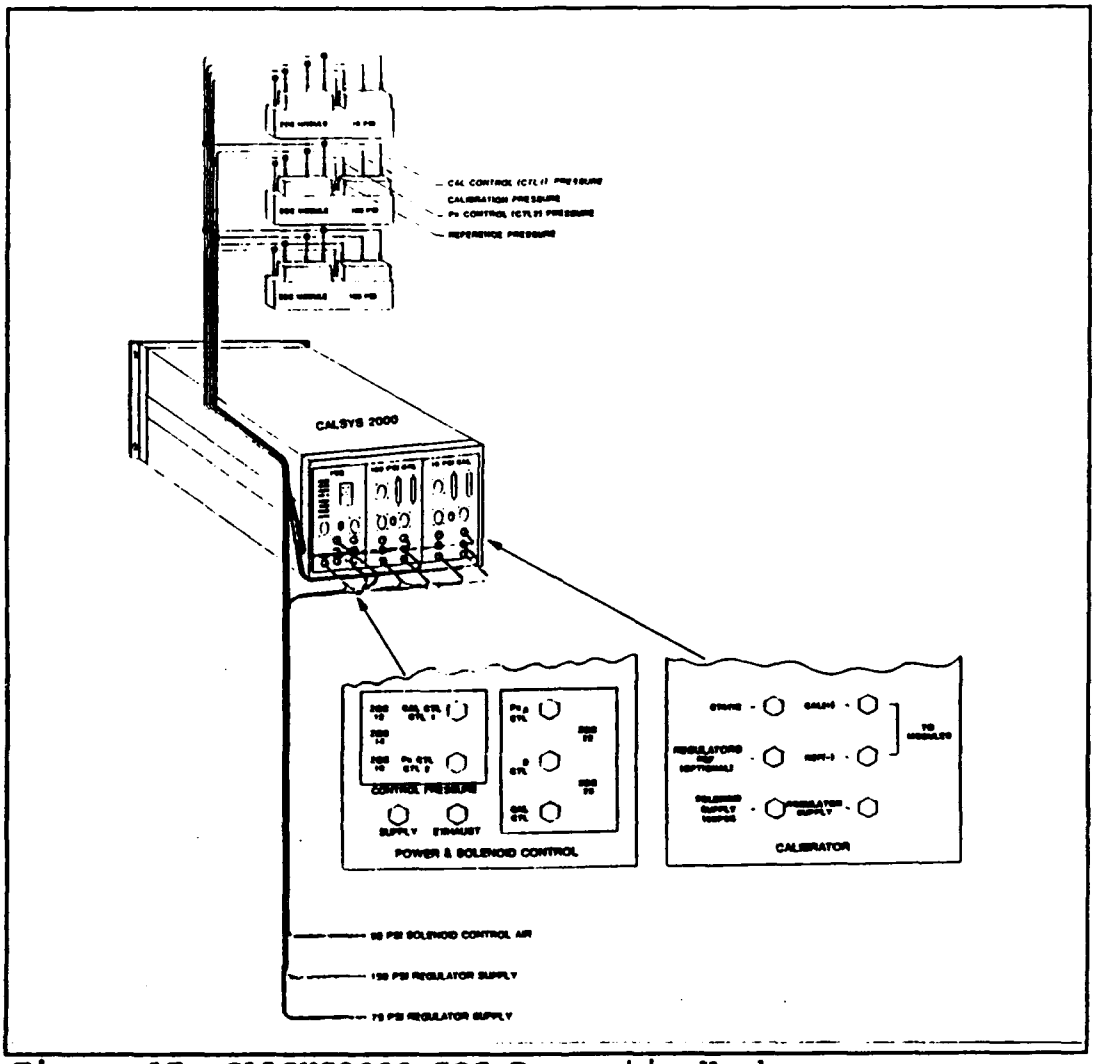


Figure 17 CALSYS2000-ZOC Pneumatic Hook-up

Figure 16 and item 4 in Figure 18). ZOC control gas to the solenoids is supplied at 90 psia from a high pressure nitrogen cylinder through the Supply port (item 7 in Figure 18).

d. CALSYS2000 Operation

The CALSYS2000 communicates with the HP9000 Computer through the RS-232C serial data link using ASCII character format commands. The commands used for the ZOC-14

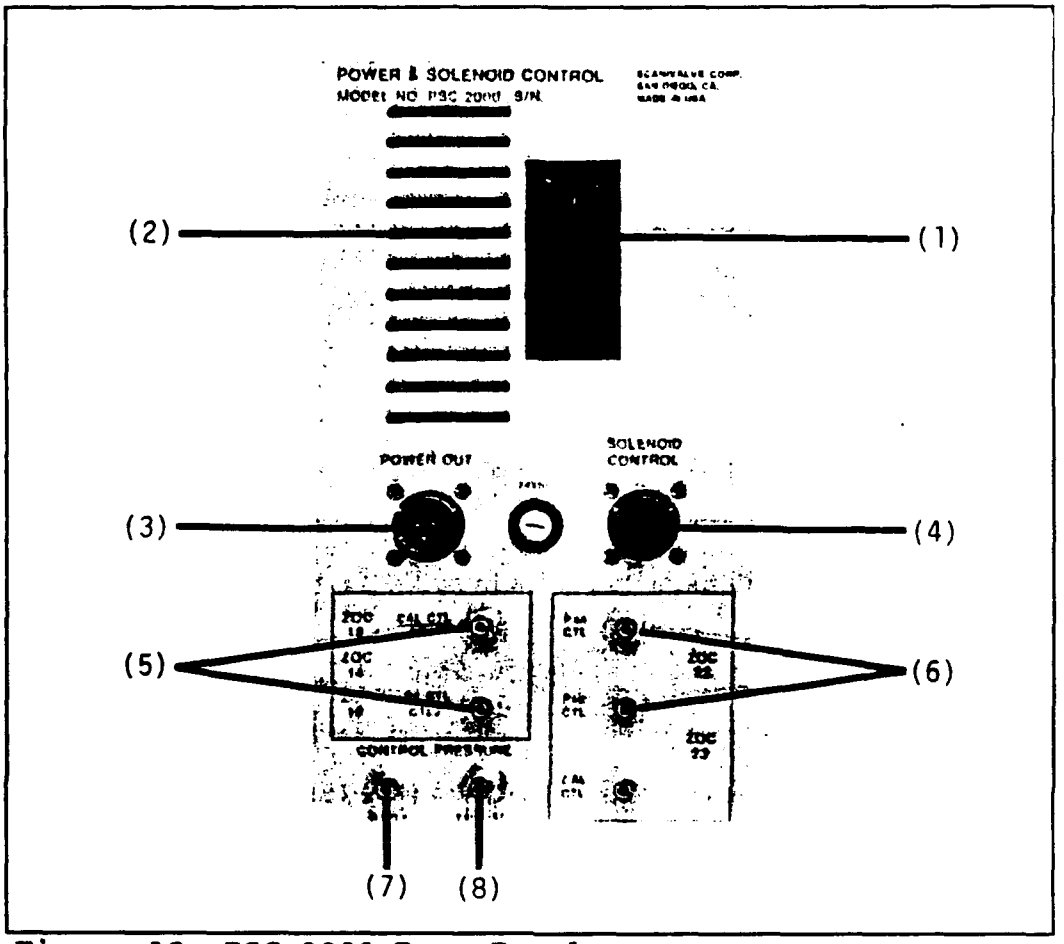


Figure 18 PSC 2000 Rear Panel

DAS are of two types:

- Commands from the HP9000 to set ZOC and CALSYS2000 modes by the actuation of control solenoids.
- Commands to provide pressure data to the HP9000, read from the Pressure Standard.

The command format for communications requires the HP9000 command statement to have the following ASCII character code format:

aCC(CR)

The "a" is the CALMOD address character which is currently "1" for the only CALMOD installed in the CALSYS2000 at the TPL. A second CALMOD would have the address character of "2". The "CC" part is the applicable command code such as "PH" which is discussed later. The "(CR)" is for a ASCII carriage return character which is "13" in decimal format.

The commands used to set ZOC and CALSYS2000 modes are summarized below:

OPERATE mode - This mode allows pressure to be measured by the ZOC by setting the valves in the ZOC to route Px air directly to the strain gage. Calibration functions are at idle. The OPERATE mode is the CALSYS2000 condition when it is first initialized or powered-up.

- aIC Initialize Calibrator to set the CALSYS2000 to its power-up condition

CALIBRATION mode - This mode is set at the ZOC. Calibration reference pressure is sent to the ZOC, and all gas is routed to the Pressure Standard for display on the CALMOD window, and made available to the HP9000.

- aPH Route high pressure gas from Regulator #1 to the ZOCs through the Calibration Manifold
- aPM Route medium pressure gas from Regulator #2 air to the ZOCs through the Calibration Manifold
- aPL Route low pressure gas from Regulator #3 to the ZOCs through the Calibration Manifold

- aZO — Gas in the Calibration and Reference Pressure Manifolds is equalized to atmopheric pressure in these manifolds
- aNL — Route low pressure gas from Regulator #3 to the ZOCs through the Reference Pressure Manifold
- aNM — Route medium pressure gas from Regulator #2 to the ZOCs through the Reference Pressure Manifold
- aNH — Route high pressure gas from Regulator #1 to the ZOCs through the Reference Pressure Manifold

Read Pressure - The CALMOD is asked for the current pressure value being sampled by the Pressure Standard. This command is used only by the ZOC-14 DAS program during the Calibration Mode. When the CALSYS2000 is in its "initialized condition", the Pressure Standard reads atmospheric pressure.

- aRP — Reads the pressure sampled by the Pressure Standard and returns the value in inches of mercury in ASCII character format: "+/-1.23456E+78 at a"

Reference 17, Chapter 4, discusses the details and available command codes use by the CALSYS2000. The above codes are only used in the ZOC-14 DAS program with the HP9000. The CALSYS2000 is designed to operate from any PC using a standard modem control program.

The HP9000 and BASIC programming language required a modification of the CALSYS2000's command format factory settings. The Line-feed (LF) ASCII character in the command line corrupted the command message to the CALSYS2000 on any subsequent command following the initial HP9000 command.

Deletion of the LF from the HP9000 command resulted in uninterrupted command of the CALSYS2000.

[In its factory configuration, the CALMOD "handshakes" with a host computer using a "prompt record" to respond to a host computer command. (See Reference 17). Following a command issued to the CALMOD, the HP9000 would receive an interrupt error and halt the program. The BASIC line commands to handle the CALMOD prompt record could not be determined. Subsequently, the factory-set prompt record "(CR)(LF)(;)" was deleted to allow the HP9000 to continue without an interrupt error. Deletion of the prompt record "(CR)(LF)(;)" was accomplished by setting the "Null" prompt record. The Null prompt record was set into the CALMOD processor using the command " SM0N " followed by " BP 1 ". The commands were communicated using a 386 PC with a modem-control program that performed computer to computer communication. The first command set the Null prompt record, the second command "burned" the Null prompt record into the CALMOD's PROM.]

e. CALSYS2000 Gas Supply System

The CALSYS2000 requires instrument quality air meeting ISA-S7.3 [Ref. 17, page 18]. The use of commercial bottled nitrogen gas more than satisfied this requirement. The supply system for the CALSYS2000 at TPL is illustrated in Figure 19. Nitrogen gas is reduced through a standard

regulator set at 90 psi, providing Solenoid Control and Regulator Air.

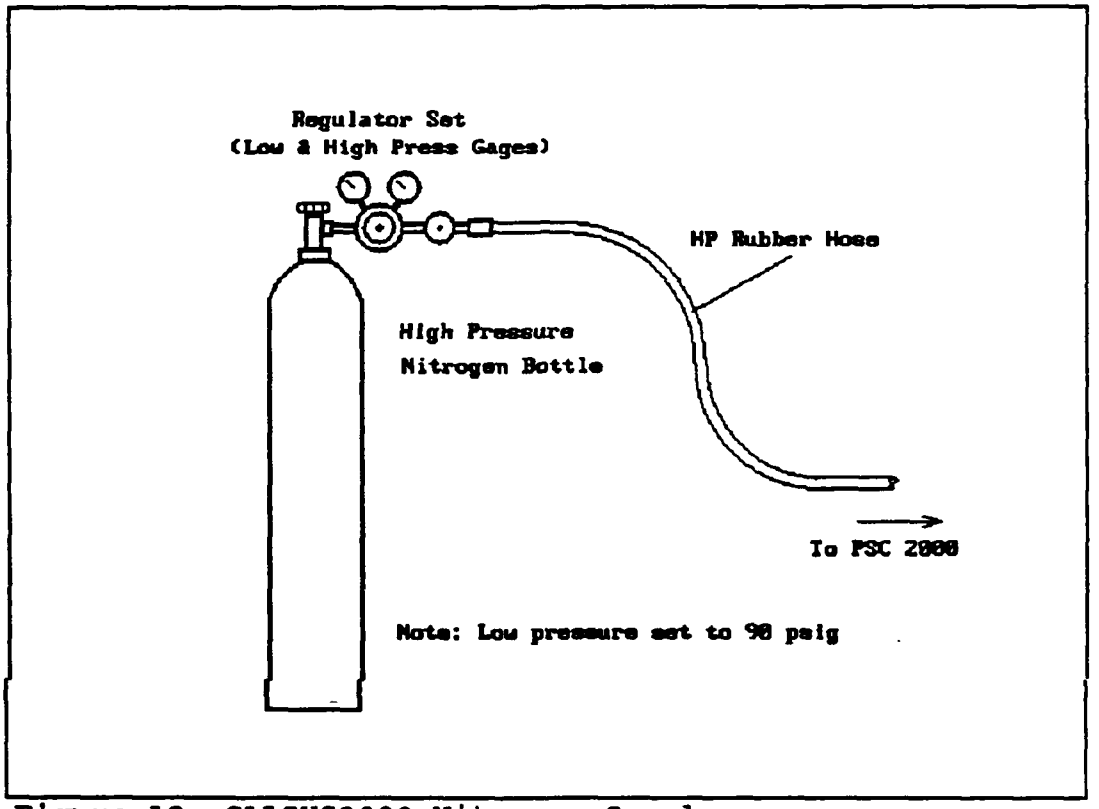


Figure 19 CALSYS2000 Nitrogen Supply

A cut-out valve (Figure 15) is installed between the PSC and CALMOD to minimize nitrogen gas consumption. The valve is set closed (handle down) when the CALSYS2000 is used in the OPERATE mode. The valve is opened for the CALIBRATE mode. Higher gas consumption occurs due to the design of the regulators in the CALMOD, which bleeds off gas to maintain a constant pressure setting.

3. Hewlett Packard Multiprogrammer (HP6944A)

a. General Description

The HP6944A (Figure 3) is a self-contained module with removable I/O cards which control various device operations or functions and interface with the HP9000. I/O cards are easily installed or removed by sliding the card into or out of an I/O slot in the enclosure. The HP9000 communicates with the I/O card through the HP6944A internal processor and Backplane Edge Connector which the installed I/O card plugs into. Electrical power supplies for the I/O cards are contained in the HP6944A enclosure. The HP6944A can support up to 16 I/O cards, depending on the cards' electrical power requirements. Figure 20 shows the HP6944A with the eight I/O cards and associated "edge plane" connecting cables used for the ZOC-14 DAS.

Reference 18 provides a detailed description of the HP6944A and a brief description of all the I/O cards available for the HP6944A. The I/O cards used in the ZOC-14 DAS are described in the following paragraphs.

b. I/O Cards

(1) 500 Khz A/D Card (HP69759A)

The A/D card provides analog to digital signal conversion from the ZOC-14 module. One A/D card is required for each ZOC. The card has a capability of performing A/D conversions at a rate of up to 500 Khz. The card is factory

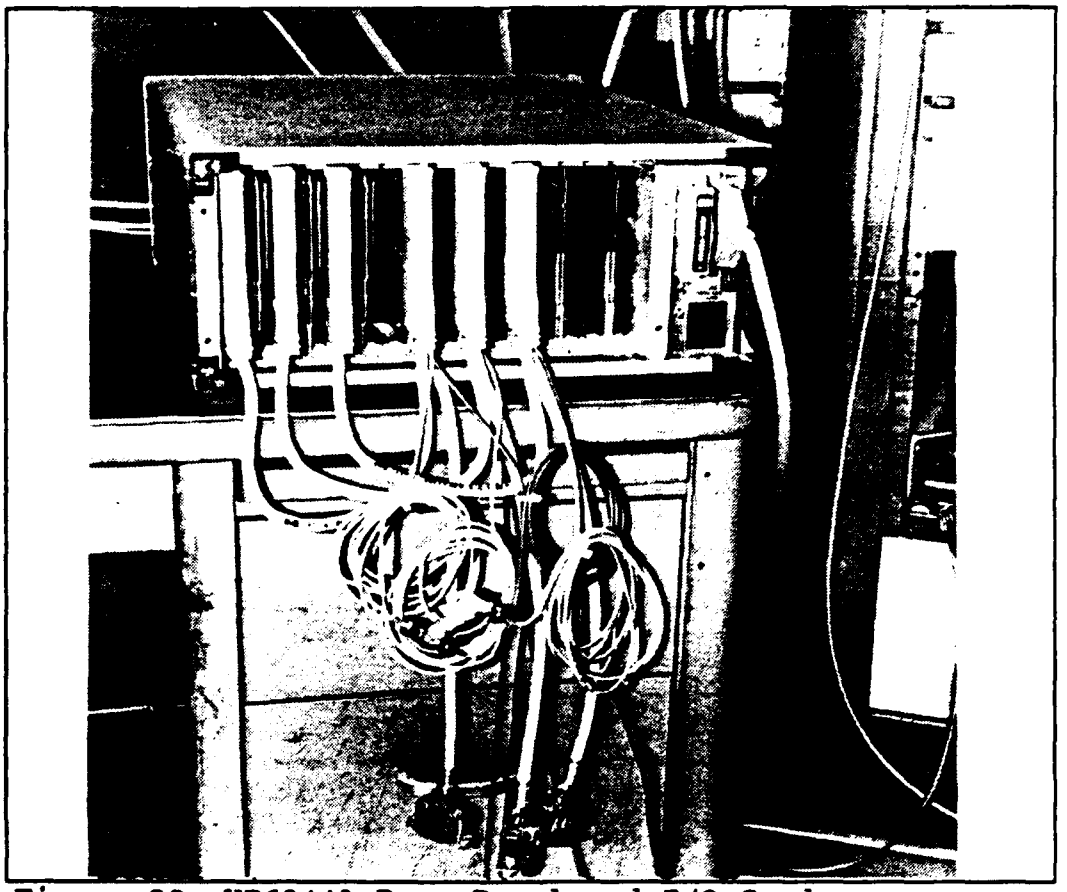


Figure 20 HP6944A Rear Panel and I/O Cards

set to handle input analog voltages between +/- 10 volts [Ref. 19]. The A/D process provides a 12 bit resolution resulting in a 5 mV digital output voltage resolution.

The HP1417A Chaining Cable (Figure 21[5]) interfaces the A/D card with the Memory Card, Pacer Card and ZOC. The digital output voltage value from the A/D card is read to the Memory Card through the HP1417A cable. The Pacer Card's trigger signal and ZOC voltage signal are received by the A/D card through Chaining Cable input leads.

---

[5] Figure 21 is in Ref. 19 as figure 2-2.

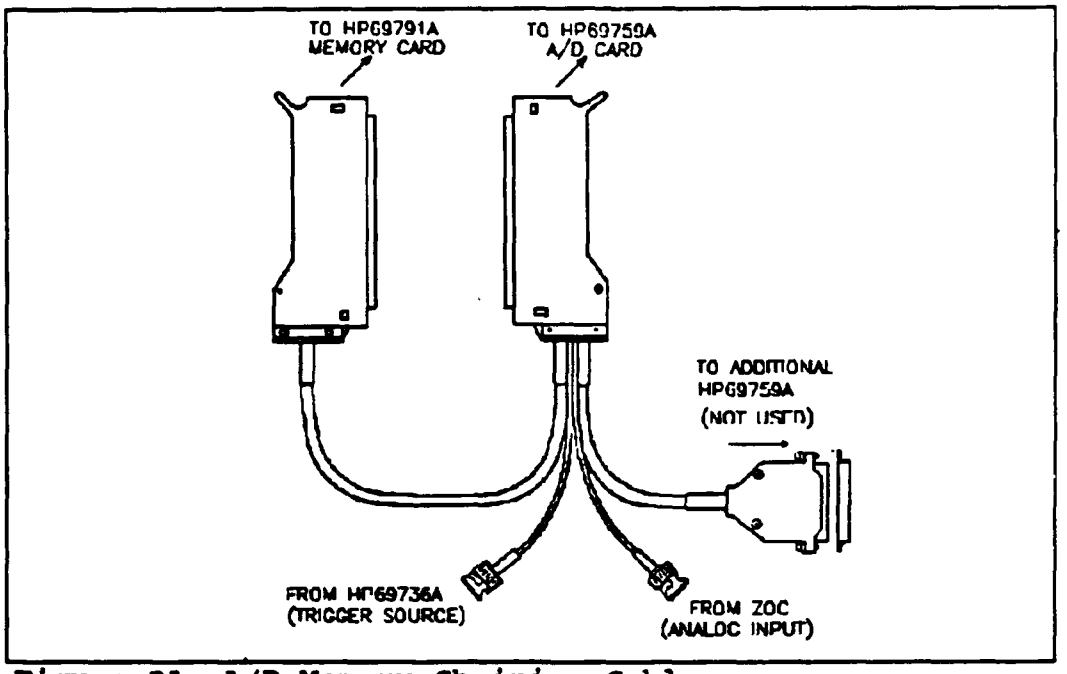


Figure 21 A/D-Memory Chaining Cable

(2) Memory Card (HP69791A)

The Memory Card is a digital data storage device used here to store ZOC voltage values from the A/D card. Data are stored in 16-bit words and the storage capacity is 65,536 words of RAM [Ref 20]. Data words can be written into the RAM at rates of up to 800 Khz. The data are extracted from the RAM into the HP9000 RAM at rates of up to 30 KHz. The difference in data rates does not limit the ZOC-14 DAS process. The DAS program is designed to collect and store all raw pressure data on the Memory Card for each acquisition cycle before any data are transferred to the HP9000.

(3) Counter/Totalizer Card (HP69775A)

The Counter Card is use in conjunction with the Timer/Pacer Card to count the number of events that occur during a specified data collection cycle. A trigger signal from the Timer/Pacer Card provides the signal pulse to step the Counter Card's integer "count-up" function. The Counter Card has two counting modes that count at rates of up to 1 Mhz [Ref. 15 and Ref. 21]. The mode used in the ZOC-14 DAS process counts from -32768 to 0. This capability allows 1023 samples per port of data, or 1023x32 events for a maximum of 32736 total data points per run.

The Counter Card provides the count value at the card's Edge Connector (Figure 22[6]) as an External Count Output. The output is a 16-bit number, which is the count value as the card is counting. Five of the sixteen bits (00 through 04) are used to drive the ZOC address value (bits A0-A4) through the Auxiliary I/O Logic Interface Device (Figure 6).

(4) Timer/Pacer Card (HP69736A)

The Pacer Card is a square wave pulse generator with a programmable pulse width feature. The pulse width is programmable down to one micro second duration or a

[6] Figure 22 is figure 3-5 in Ref. 21.

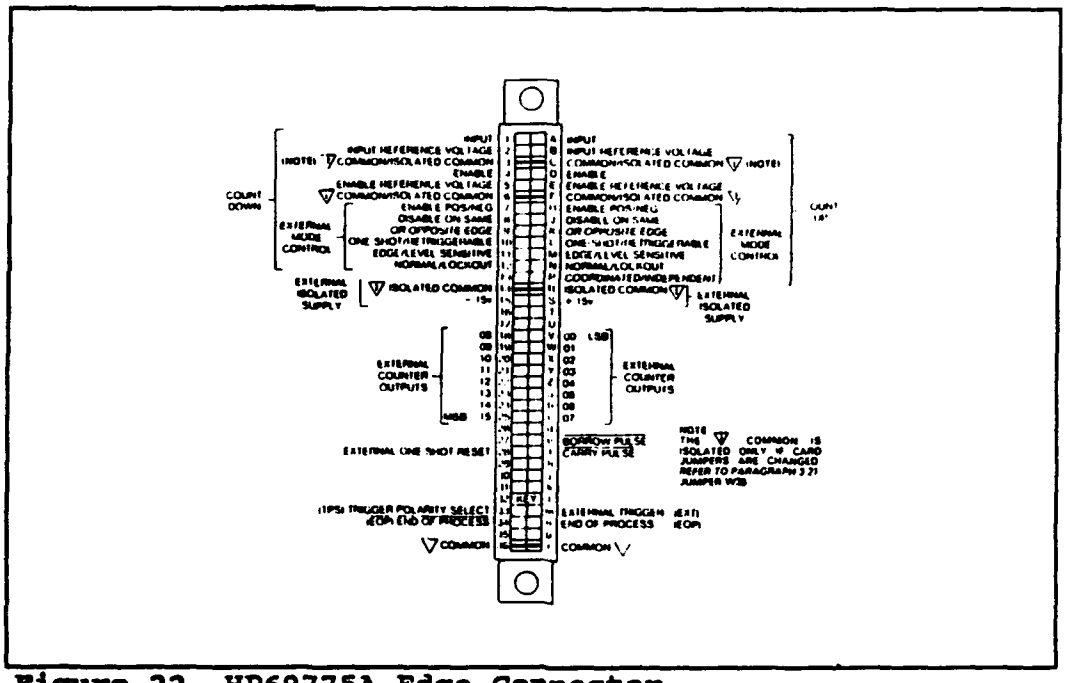


Figure 22 HP69775A Edge Connector

one MHz square wave[7]. The Pacer Card provides the trigger for the A/D Card and timing step trigger for the Counter Card.

c. Multiprogrammer Configuration

(1) ZOC-14 Integration

The ZOC address format and the Counter Card's External Counter Output are the common link that interface the two devices. The Counter Card's count-up feature is used to control the ZOC port address. Starting at "-32", for which the binary equivalent is "00000" at the card edge, the card counts up to zero. Since 00000 is the binary number which is required to set the ZOC address to port number "1", the

---

[7] Refer to Ref. 15 and Ref. 22 for details on the HP69736A.

counting process steps the ZOC from port 1 to port 32. The count-up process realizes the following sequence:

decimal count	binary count	ZOC port address
-32	00000	01
-31	00001	02
-29	00010	03
	. . .	
-02	11110	31
-01	11111	32

Initiating the count-up from "-64" results in two scanning passes through the 32 port ZOC-14 since only the last five bits are identified in the ZOC address. Selection of the count-up value determines the number of scans through all the ZOC's ports.

The Counter Card's external output is set to 0.0-0.5 volt for "logic low" and to 2.0-5.0 volt for "logic high" [Ref. 21, page 1-2]. The ZOC address bits (A0-A4 in Figure 10) require an open collector Transistor-Transistor-Logic (TTL) driver to provide a ground "low" and an open "high" signal. The Auxiliary I/O Logic Interface Device (Figure 23) was designed to interface the HP69775A with the ZOC. Figure 24 is the electrical schematic of the device. The TTL 7404 is powered by a 5 volt regulator [Ref. 23]. The regulator receives 15 VDC power from the Counter Card's external edge connector.

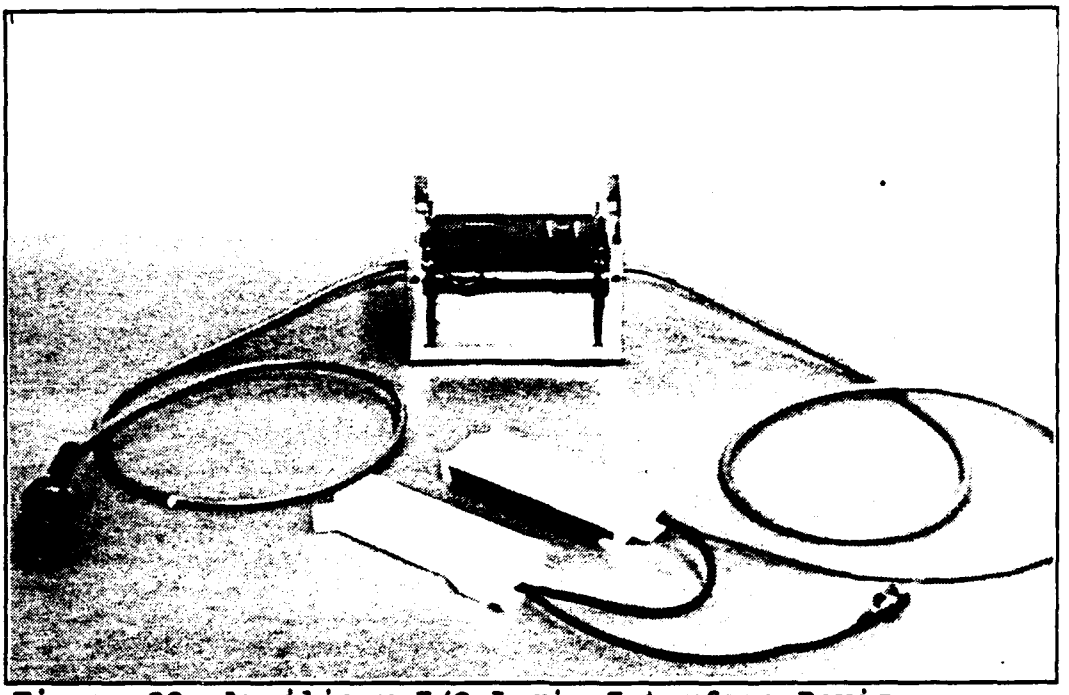


Figure 23 Auxiliary I/O Logic Interface Device

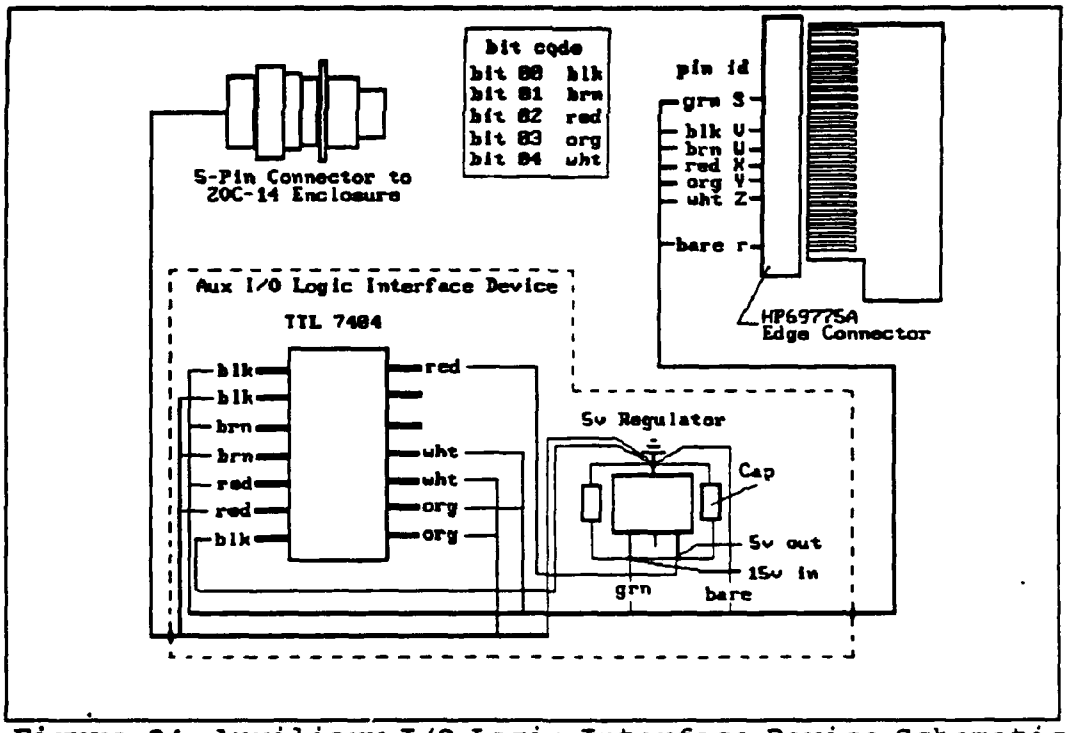


Figure 24 Auxiliary I/O Logic Interface Device Schematic

(2) I/O Card Slot Configuration

The current configuration of the I/O cards and HP6944A provides the capability to collect pressure data from up to three ZOC-14 modules. Figure 25 illustrates the I/O card configuration and wiring to support the full ZOC-14 DAS.

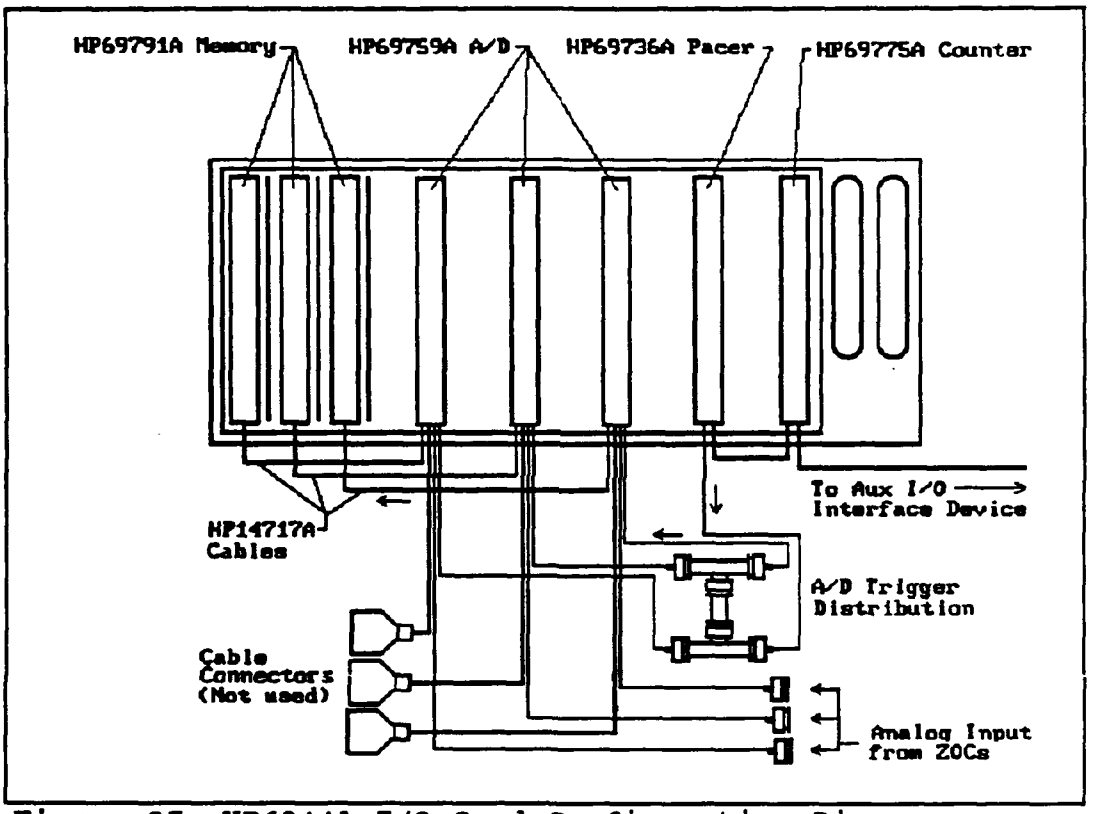


Figure 25 HP6944A I/O Card Configuration Diagram

Figure 26 shows the ZOC-14 DAS (not including the CALSYS2000) with one ZOC-14 module and all electrical lines connected.

The HP6944A has a power limitation determined by the built-in power supplies. Table I illustrates the HP6944A power supply availability and the I/O card power requirements. The HP6944A provides one 5 volt, one 12 volt and three 18 volt power supplies. The maximum amperage

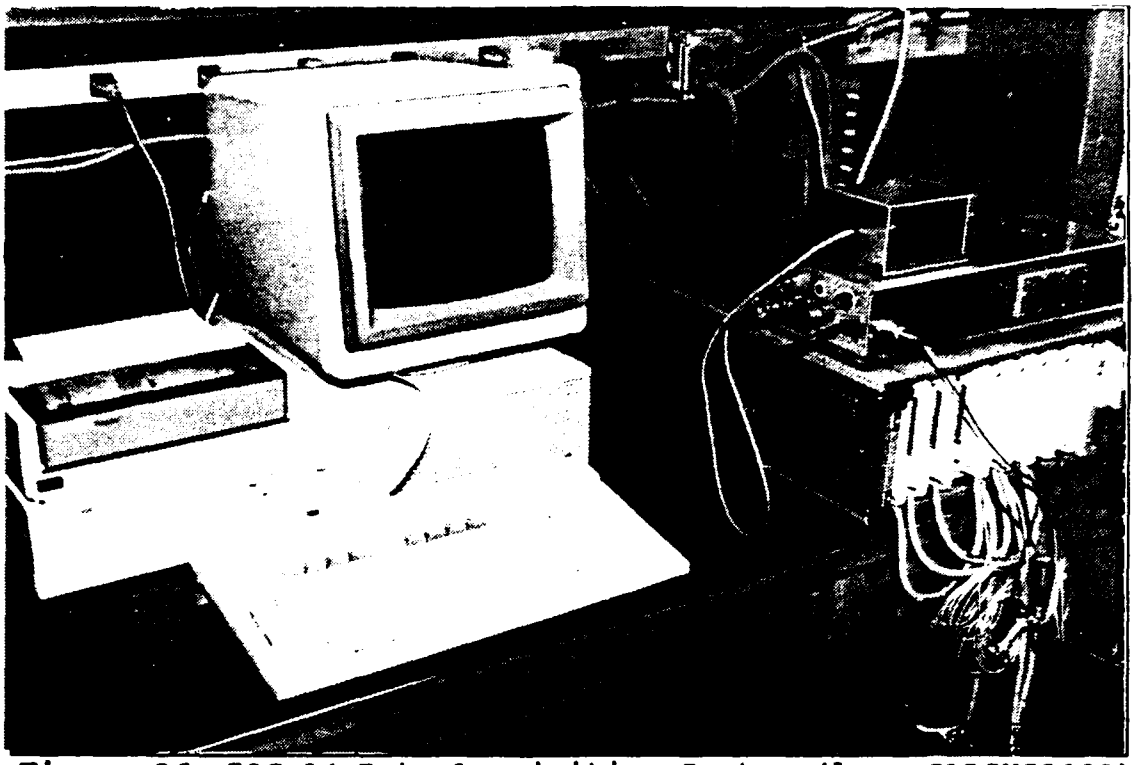


Figure 26 ZOC-14 Data Acquisition System (less CALSYS2000)

ratings are shown in Table I. The power required by the I/O cards are shown in amperes. The upper value in the table is the rating per card. For the HP69791A and HP69759A cards, the second value is the total rating for three cards of each type (which are required for three ZOC modules). The total amperage required for the three HP69759A cards exceeded the capacity of a single 18 volt power supply. Therefore, the first HP69759A card receives its power from the first 18 volt power supply. The second and third HP69759A cards receive their power from the second 18 volt power supply, from which they draw a total of 0.8 amperes. The HP69759A cards were set by the factory to draw power from the first 18 volt power

Table I HP6944A Power Supply Allocation

Device	Power Supplies				
HP6944A	+5V 16.0A (1)	+12V 2.0A (1)	-12V 2.0A (1)	+18V 0.85A (3)	-18V 0.4A (3)
Cards	Power Required				
HP69791A	3.6A 10.8A				
HP69759A HP69759A HP69759A	0.7A 0.7A 0.7A			(1)0.4A (2)0.4A (2)0.4A	(1)0.075A (2)0.075A (2)0.075A
HP69736A	0.75A				
HP69775A	0.75A	0.12A	0.15A	(1)0.12A	(1)0.150A
TOTAL	14.4A	0.12A	0.15A	(1)0.52A (2)0.80A	(1)0.225A (2)0.150A

supply. Access to the second 18 volt power supply required alterations to be made to the Power Supply Jumpers for the second and third HP69759A cards [Ref. 19, page 3-1].

C. ZOC-14 DAS Software Description

1. General Overview

Hewlett Packard's BASIC 5.13 program language is utilized by the HP9000. Code is written in this language to communicate with the various data acquisition devices, process data and store the data to disk, and to output results on a printer or plotter. The HP6944A, a central component of the ZOC-14 DAS, is a relatively complex device which can perform numerous functions depending on how it is configured. The use of the HP6944A here required the generation of a unique software package in BASIC that integrated the HP6944A internal processor with the configuration of I/O cards selected here. The HP14753A Computer Aided Test (CAT) Programming Package provided the means to generate the software for that interface.

2. Data Acquisition Program

a. HP14753A CAT Program Package

The HP14753A CAT Programming Package (Figure 27)[Ref. 15] is software (and documentation) required to operate the HP6944A and the associated I/O cards. The programmer uses the Hewlett Packard provided skeleton CAT example program to develop a tailored application program to perform specific processes. The ZOC-14 DAS application program, was developed using the CAT Programming Package, to

Figure 27 HP14753A CAT Programming Package

perform ZOC data collection, reduction, and storage titled "SCAN_ZOC_05" (Appendix A, Figure A17).

The CAT programming package provides the capability of integrating selected I/O cards into a unique function that combines the cards' capabilities into one function. Two particular applications used here combined the A/D and Memory Cards into a "Buffer" function, and the Pacer and Counter Cards into a "Timer" function.

The Buffer function performs the task of collecting analog data at a collection rate controlled by an external trigger device, converts the analog data to digital format, and temporarily stores the data. The Buffer function allows the high speed data collection rate of 500 KHz, which is only limited by the A/D conversion rate. Figure 28 is a

schematic of the wiring that links the A/D and Memory Cards together through the HP1417A Chaining Cable (Figure 21).

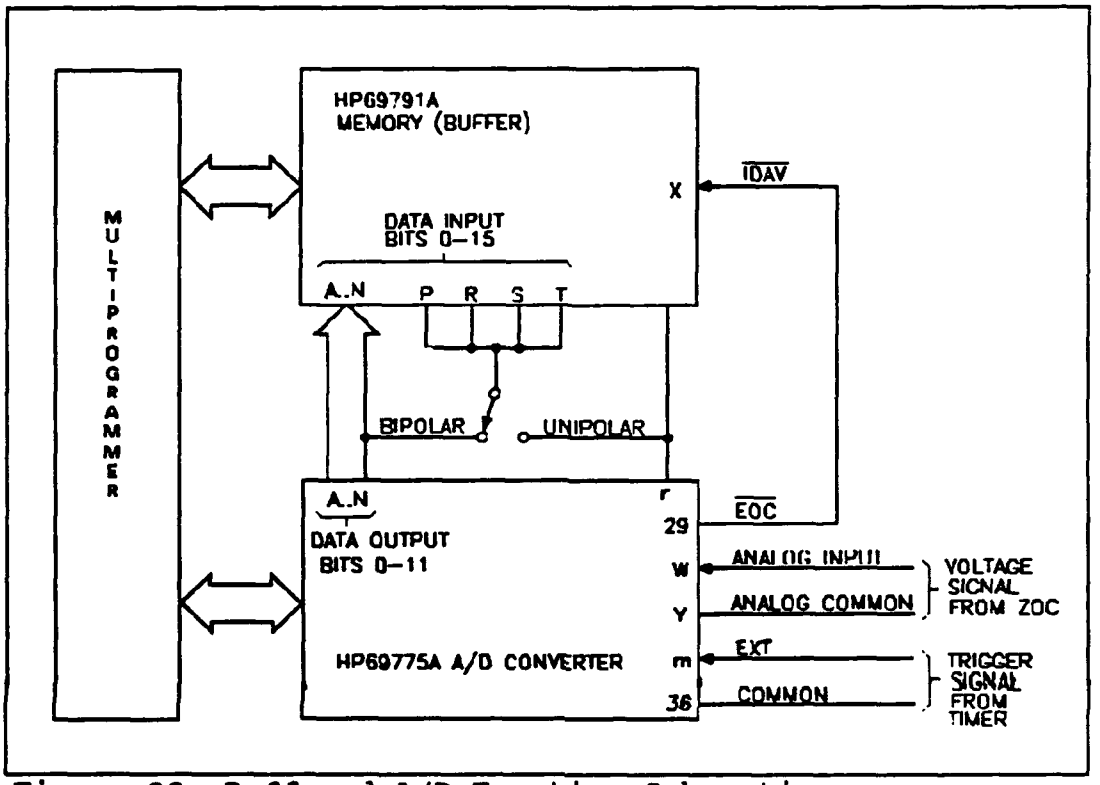


Figure 28 Buffered A/D Function Schematic

The Timer function provides a square wave trigger pulse of a specified pulse width and a specified number of repetitions. The pulse width determines the data collection rate. In the present application, the repetition number divided by 32, the number of ports on the ZOC-14, determines the number of scans the ZOCs undergo. The repetition number is therefore always a multiple of 32. Figure 29 is the schematic of the wiring that links the Counter and Pacer Cards together. The edge connectors and cables which were made to implement the wiring shown in Figure 29, can be seen in

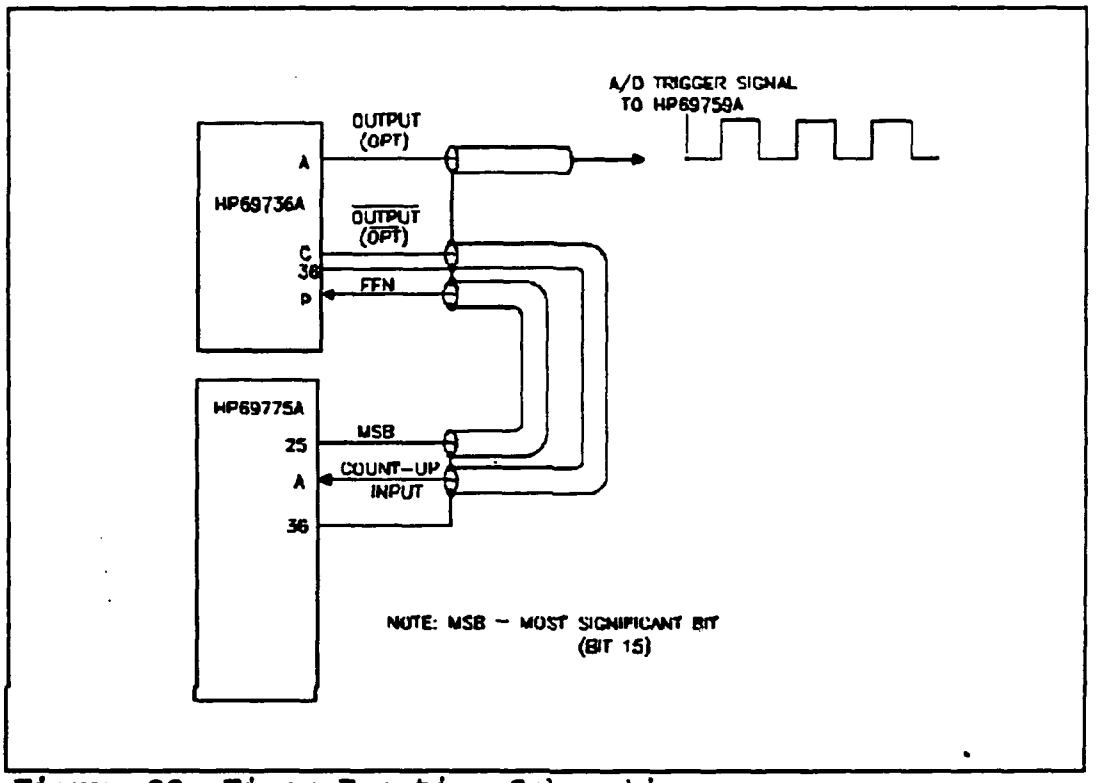


Figure 29 Timer Function Schematic

Figure 23.

The CAT programming process requires identification of the installed I/O cards and their initial function settings through a user-specified "Configuration File". The file provides the flexibility of writing a general application program that works readily with various configurations of the HP6944A(s). Each configuration file is unique to each HP6944A hardware configuration[8].

[8] The Configuration File must accurately reflect the HP6944A(s) I/O card configuration or a software error occurs.

Print-outs of the ZOC-14 DAS program Configuration File are listed in Figure A1[9]. [Note that the names listed in Figure A1 are variables used by the program SCAN_ZOC_05.].

b. DAS Program Design

(1) Program Design Features

The ZOC-14 DAS Program SCAN_ZOC_05 incorporated several design features intended to obtain a user-friendly data collection program. The program

- has full control of the ZOCs and CALSYS2000 as interfaced with the HP9000 and HP6944A
- utilizes the Zero Operate Calibrate principle for data collection from the ZOCs
- uses keyboard function keys to allow selection of various operations within the program
- creates and identifies data files automatically based on the type of data, date, and run number
- offers data file storage on hard and floppy disk drives.

(2) Data Files

Three different data files are used when acquiring data from each ZOC-14. Implementation of the Zero, Operate and Calibrate procedure required the data acquisition process to

- collect and store raw pressure data from the ZOCs

[9] Reference 15 provides details on Configuration File creation and alterations.

- collect and store calibration pressure data from the ZOCs and corresponding applied pressure standard data from the CALMOD 2000
- reduce the raw pressure date using calibration curves and store the reduced pressure data.

The program uses BDAT type files with Integer and Real number formats[10]. The raw data file is in Integer (2-byte) format, with each record corresponding to an output voltage value.

The calibration file uses a Real number (8-byte) format storing data from a 33 row by 11 column array[11]. Each row (1-32) corresponds to a ZOC Port. The Zero (0) row contains ZOC-specific parameters. Columns 4-10 in the Zero row contains pressure values measured by the CALMOD Pressure Standard for each of the seven calibration settings. Columns 4-10 (rows 1-32) contain voltages output by the ZOC for each of the calibration pressures applied through the CAL and REF manifolds. Columns 0-3 (rows 1-32) contain calibration curve fit coefficients for a third order polynomial. These coefficients are derived using the Least Squares numerical curve fit method using the recorded voltages (rows 1-32) plotted against the applied (standard) calibration pressures (row zero). Each row in the array corresponds to a record in the calibration data file consisting of 8x33 bytes.

---

[10] Refer to Ref. 6, chp 7, for details and advantages of BDAT format files compared to ASCII files.

[11] Refer to Figure A17, lines 2350 to 2440.

The reduced data file uses a Real number (8-byte) format, storing data from a n-row by 33 column array. Each row corresponds to one scan of a ZOC. The first column (0) stores the nth-scan number. Columns 1-32 correspond to the output of each ZOC port converted to pressure units (inches of mercury). Each array row occupies a record of 8x33 bytes in the reduced data file.

Figure 30 provides a sample listing of the data files created in acquiring data using three (3) ZOCs in one (1) acquisition cycle.

```
:,700,0,1
VOLUME LABEL: DATA
FILE NAME PRO TYPE  REC/FILE BYTE/REC  ADDRESS     DATE      TIME

ZW1205161 BDAT         97        2        592   16-May-92 14:54
ZW2205161 BDAT         97        2        594   16-May-92 14:54
ZW3205161 BDAT         97        2        596   16-May-92 14:54
ZC1205161 BDAT         33       88        598   16-May-92 14:55
ZC2205161 BDAT         33       88        611   16-May-92 14:55
ZC3205161 BDAT         33       88        624   16-May-92 14:55
ZR1205161 BDAT          3      264        637   16-May-92 14:55
ZR2205161 BDAT          3      264        642   16-May-92 14:55
ZR3205161 BDAT          3      264        647   16-May-92 14:55
```

Figure 30 ZOC-14 DAS Data File Listing

Each data file name uses the format,

<ZW><ZOC #><Date><Run #>

"ZW" identifies raw data files. ("ZC" identifies calibration data files, and "ZR" identifies reduced data files). The ZOC # is for ZOCs 1-3. The format for the date is YMMDD. The Run # has values 1-99. Hence, for example, the file ZR1205161 holds reduced data for ZOC #1 collected on May 16, 1992 during

Run #1. The REC/FILE column shows three records, identifying that three scans of data for the 32 ports were collected. The BYTE/REC column identifies 264 bytes for each record (scan), corresponding to the 8x33 bytes for the row format addressed in the previous paragraph.

(3) Program Functional Flow Process

The program SCAN_ZOC_05 follows the functional flow process illustrated in Figure A2 through Figure A16 in Appendix A. The program relies on extensive use of subroutines to maintain program architectural simplicity through use of the CALL statement. The option to use one to three ZOCs requires only the repetitious use of the different subroutines.

Function keys are used for program control. The main program is segmented into blocks. Each block or set of blocks is initiated by a function key (ie. f1 through f8). The "GOTO Hold" statement executes a continuous loop sequence which is only interrupted by a function key selection, thus providing the mechanism for program control.

The storing of raw pressure data and calibration data onto the hard drive allows repeated data collection runs. The data reduction routine is function key selected and can be performed after each data collection run, or at any later time. The program prompts the user for data

file name information when data reduction is not selected immediately following a data collection run.

c. DAS Program Use

(1) HP9000 Operation and Rudimentary Commands

The HP9000 is the controlling unit for the ZOC-14 DAS. Its operation is similar to that of a desk top computer system with no complex initialization or "boot-up" process.

The HP9000 has its own built-in boot-up routine as an integral part of BASIC [Ref. 4]. The boot-up process addresses the "SYSB51.HP-UX" file on the hard drive's root directory for loading various function drivers called "Binaries". Integration of the HP14753A CAT programming package in the present work required modification to the SYSB51 file to provide sufficient RAM space to operate the CAT programs[12].

The boot-up process is initiated when power is applied to the HP9000. The CRT begins displaying a series of screens during the boot-up. With the modification which was made to the SYSB51 file, completion of the boot-up process now gives the screen display shown in Figure 31, which is generated by the AUTOST program file. This screen, referred

[12] Refer to Ref. 4, chp 5 and Ref. 15, pg. 1-5, for details on the SYSB51 file modifications.

to as the "Main Menu", provides a function key selection menu to access the operations available on the HP9000.

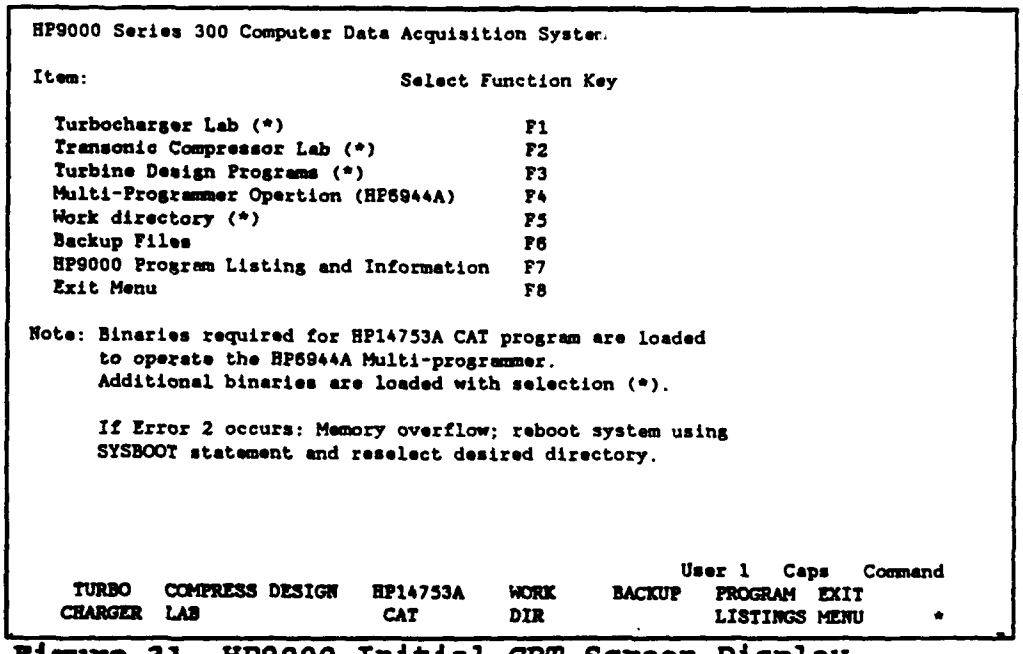


Figure 31 HP9000 Initial CRT Screen Display

Selection of the "F8" function key clears the screen and places the user in the root directory. At any time when the term "idle" is displayed in the lower right corner of the CRT screen, there are no programs in execution, and the user can make keyboard command entries.

There are five essential, rudimentary commands that the user needs to know to use BASIC[13]. These commands are LOAD, RUN, CAT, MSI, and RE-S ORE. The LOAD command will load the program which is named between quotes into the HP9000 RAM. RUN will execute the program currently in RAM. CAT will

[13] It is recommended that the user reviews Ref. 13 and 14 for an operational understanding of BASIC.

list to the CRT screen all the files on the current storage drive and directory. MSI ("MASS STORAGE IS") will select a directory and/or drive to be what is named between quotes. The RE-STORE command copies the program in the RAM to the file on the currently named MSI. Caution must be applied when using the RE-STORE command. This command will write over any previously existing program with the same name, resulting in the loss of the previous program.

Return to the Main Menu is possible from any directory on the hard drive. Located on the root directory of the hard drive is a program called "RETURN_MAIN". Load the program by typing the following commands:

- LOAD "/RETURN_MAIN", followed by the <Enter> key
- RUN, followed by the <Enter> key

The "/" character identifies the root directory. The above two steps apply for loading and executing any program.

(2) Operating the ZOC-14 DAS Program

The ZOC-14 DAS Program uses function keys and selective keyboard data entries. The following steps should be followed to operate the ZOC-14 DAS.

1. Main Menu: Depress function key F4 (Figure 31) to call-up the HP Multi-programmer Operating Menu.
2. HP6944A Operation Menu: Depress function key F1 to call-up the ZOC-14 Modules Menu.

3. Zoc Electronic Pressure Module Operation Menu: Figure 32 is displayed.

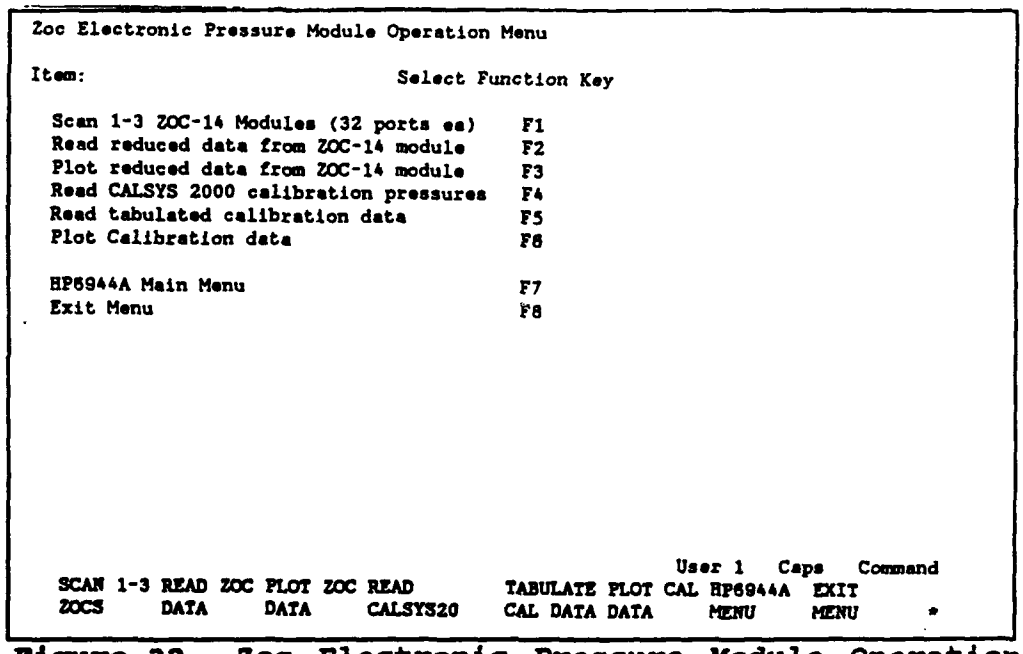


Figure 32 Zoc Electronic Pressure Module Operation Menu

4. **Check: Ensure that the CALSYS2000, ZOC Enclosure and HP6944A are properly connected and energized.**

5. **Check: Ensure that the nitrogen gas supply is connected to the CALSYS2000 and about 90 psi is set on the regulator.**

6. **CALSYS2000 Regulators: Set the High, Medium, and Low Pressure regulators on the CALSYS2000. The three pressure values should be evenly distributed within the span of the ZOC's positive pressure rated range. Precise settings are not critical. Since 50 psid and 15 psid ZOCs are used presently, set the regulators between 0-30.6 inches of mercury to avoid over pressurizing the 15 psid ZOCs.**

7. CALSYS2000 Verification: Select function key F4 (Figure 32) to cycle the CALSYS2000 and verify the pressure settings. (Note: This should be done whenever the CALSYS2000 is first energized to clear the RS-232C of noise). Completion of this step returns the HP9000 to the ZOC operation menu.

8. Load SCAN_ZOC_05: Select function key F1 (SCAN ZOCS) from the ZOC operation menu to initiate ZOC scanning program SCAN_ZOC_05.

9. Introduction: A series of screen displays occur while SCAN_ZOC_05 is loading. The "Introduction" screen (Figure 33) is displayed indicating that the program is waiting for a function key input.

```
Introduction.  Program SCAN_ZOC_05:

  - Scans 1-3 Zoc-14 Modules simultaneously (32 pressure sensing ports each).
  - Uses Zero Operate Calibrate (ZOC) principal:
      - Collects raw pressure data (Zero Operate)
      - Collects calibration data (Calibrate)
      - Reduces and stores data on selected hard or floppy drive.
  - CALSYS2000 Calibration Module used for the reference pressure standard.
  - Raw pressure data reduced using calibration data from CALSYS2000
    and Zocs in the calibration mode.

Input variables: Hard and Floppy drive for data storage
                 Sample frequency per port (1-50,000 Hz)
                 Samples per Port (1-1021)
                 Number of Zocs and their capacity

Output files:    Raw data =>      ZW(Zoc#)(Date YMMDD)(Run#)
                 Calibration =>   ZC(Zoc#)(Date YMMDD)(Run#)
                 Reduced data =>  ZR(Zoc#)(Date YMMDD)(Run#)

Select F2 key for Key Menu, F3 for system inputs, or F8 for data reduction.


                                                    User 1   Caps   Command
  Intro    Key      Set-up   Data        Collect  Reduce   List     Exit
           Menu               Preps       Data     Data     Copy              *
```

Figure 33 SCAN_ZOC_05 Introduction Screen

10. Set-up Selection: Select function key F3 (Set-up) to initialize the program. The "Set-up" screen will be displayed. (Note, selection of F4 or F5 at this time results in an error message and asks for re-selection.)

11. Set-up Inputs: The user will be prompted for the designated data storage drive (select :,700,0,1 if equipped)[14], data acquisition rate, the number of

[14] The program operates on multi-partitioned hard drives. TPL's HP9153C Disk Drive has two partitions. Drive :,700 is the main drive and is in HFS format with 10 Mbytes of space. Drive :,700,0,1 is the second "DATA" drive and is in LIF format with 30 Mbytes of space. LIF format has faster data transfer rates.

samples per port (scans) of data to be taken, and the number of ZOCs to be used. For each ZOC, the user will be prompted for the CALMOD to be assigned to the ZOC. Currently only one CALMOD is installed. Enter 1 for all three ZOCs. When two CALMODs are installed, enter 1 or 2 to the applicable ZOC according to the CALMOD regulator settings.

12. Set-up Display: The System Set-up screen (Figure 34) is updated as entries are made. The program searches the data drive for data files and creates the next sequential data file name for the current date.

```
System Set-up.

Data acquisition rate:                          10000  Hz
Number of samples per port:                     3
Number of Zocs to be scanned:                   3
Total raw data acquisition time:                .0093  sec.
Total calibration data acquisition time:        10.5155  sec.

Data storage disc =>:,700,0,1
Data will be stored in the following files beginning with Run # 1

Raw data file:          ZW1204291
Calibration data file:  ZC1204291
Reduced data file:      ZR1204291

Raw data file:          ZW2204291
Calibration data file:  ZC2204291
Reduced data file:      ZR2204291

Raw data file:          ZW3204291
Calibration data file:  ZC3204291
Reduced data file:      ZR3204291
```

Figure 34 SCAN_ZOC_05 System Set-up Screen

13. Data Collection Preparations: Select function key F4 (Data Preps) for data collection preparations (Figure 35).

14. Collect Data: Select function key F5 (Collect data) to begin data collection. The CRT will display the results of the collection as illustrated in Figure 36.

15. Data Reduction: To reduce data, select function key F6 (Reduce Data). To make changes to the Set-up, select function key F3. To conduct another raw data collection run, select function key F4. To exit the

program, select function key F8. Selection of F6 results in the display illustrated in Figure 37.

16. List/Copy Files: Selection of function key F7 (List Files) will list all current data files on the storage drive as illustrated in Figure 38. The user

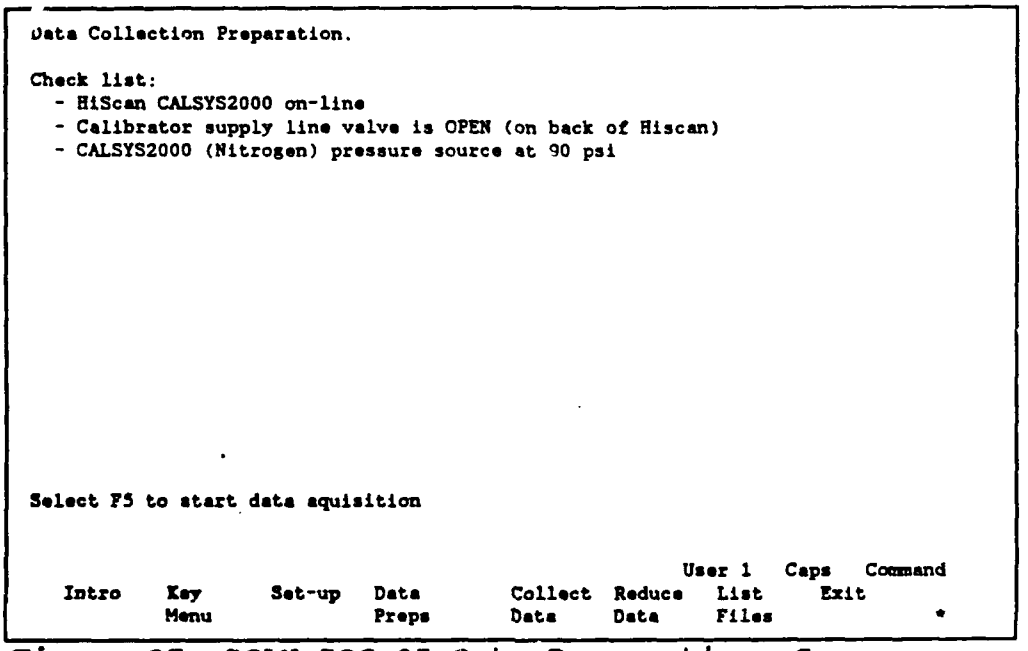


Figure 35 SCAN_ZOC_05 Data Preparations Screen

will be prompted if he wants to store all the listed files to the floppy drive :,700,1. Selection of "Yes" results in the over-writing of the old files with the same file name.

17. Exit: Select function key F8 (Exit) to exit the SCAN_ZOC_05 program and return to the ZOC operation menu (Figure 32).

3. Data Analysis and Auxiliary ZOC-14 Programs

a. Utility Programs

The ZOC operation menu (Figure 32) displays several utility programs which were written for system analysis and

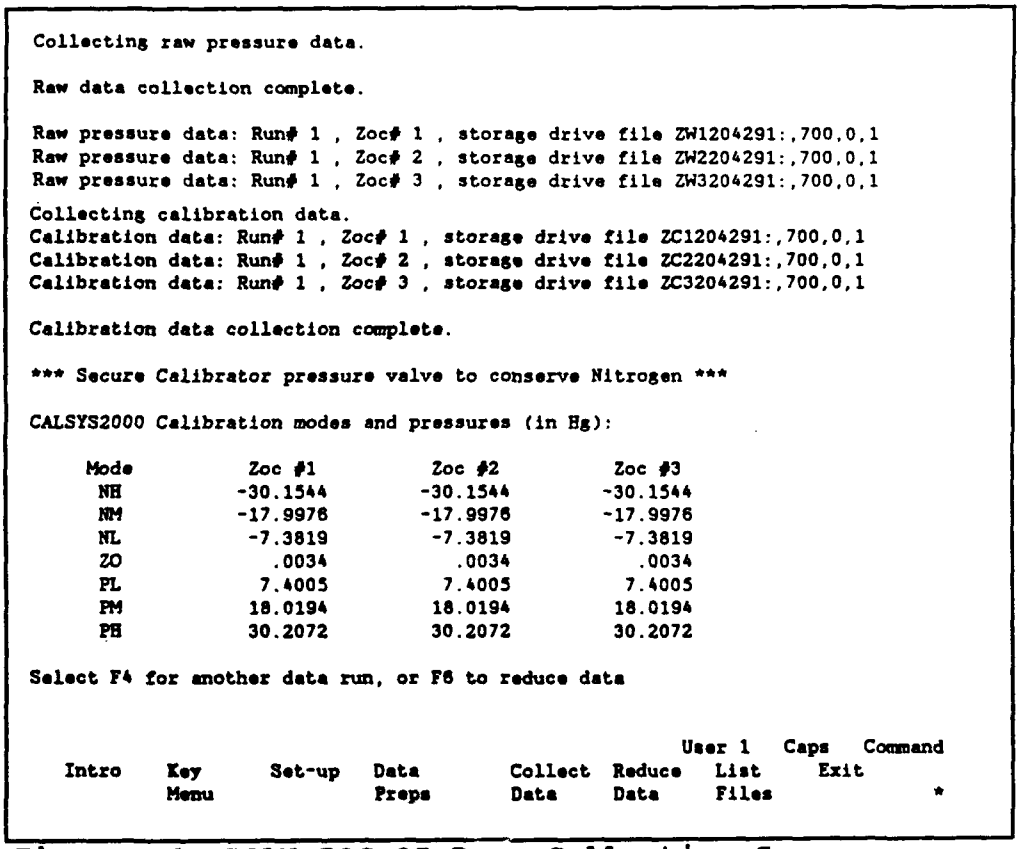


Figure 36 SCAN_ZOC_05 Data Collection Screen

data display. These utility programs, listed in Appendix A, are tailored for use with the SCAN_ZOC_05 data file formats[15].

b. ZOC-14 Utility Program Application Examples

An air source regulated to 30.0 inches of mercury gauge was used to verify the acquisition hardware and software. Application programs were written to analyse the results.

---

[15] Data files created from previous versions of the SCAN_ZOC programs listed in Appendix B are not compatible with the listed utility programs.

```
Calibration and Raw data reduction and storage.

Current files on storage disc :,700,0,1 for date 20429

   ZW1204291   ZC1204291   -
   ZW2204291   ZC2204291   -
   ZW3204291   ZC3204291   -

Data reduction: Run# 1 , Zoc# 1
Calibration data reduced and transferred to ZC1204291
Raw data reduced and transferred to ZR1204291

Data reduction: Run# 1 , Zoc# 2
Calibration data reduced and transferred to ZC2204291
Raw data reduced and transferred to ZR2204291

Data reduction: Run# 1 , Zoc# 3
Calibration data reduced and transferred to ZC3204291
Raw data reduced and transferred to ZR3204291

Select F3 reinitialize set-up for data collection, or F8 to Exit


                                                    User 1          Command
   Intro    Key      Set-up   Data       Collect  Reduce   List     Exit
            Menu               Preps      Data     Data     Files             *
```

Figure 37 SCAN_ZOC_05 Data Reduction Screen

```
List Raw, Calibration and Reduced data files.

Data storage drive name =>  :,700,0,1

Current files on storage disc for date 20429

   ZW1204291   ZC1204291   ZR1204291

   ZW2204291   ZC2204291   ZR2204291

   ZW3204291   ZC3204291   ZR3204291




Select F2 to return to menu, or F8 to Exit


                                                    User 1          Command
   Intro    Key      Set-up   Data       Collect  Reduce   List     Exit
            Menu               Preps      Data     Data     Files             *
```

Figure 38 SCAN_ZOC_05 List Files Screen

The program "READ_ZOC" (Figure A18) is selected by function key F2 (READ DATA). The program's results are illustrated in Figure 39 for ZOC #3 (rated at 15 psid). The

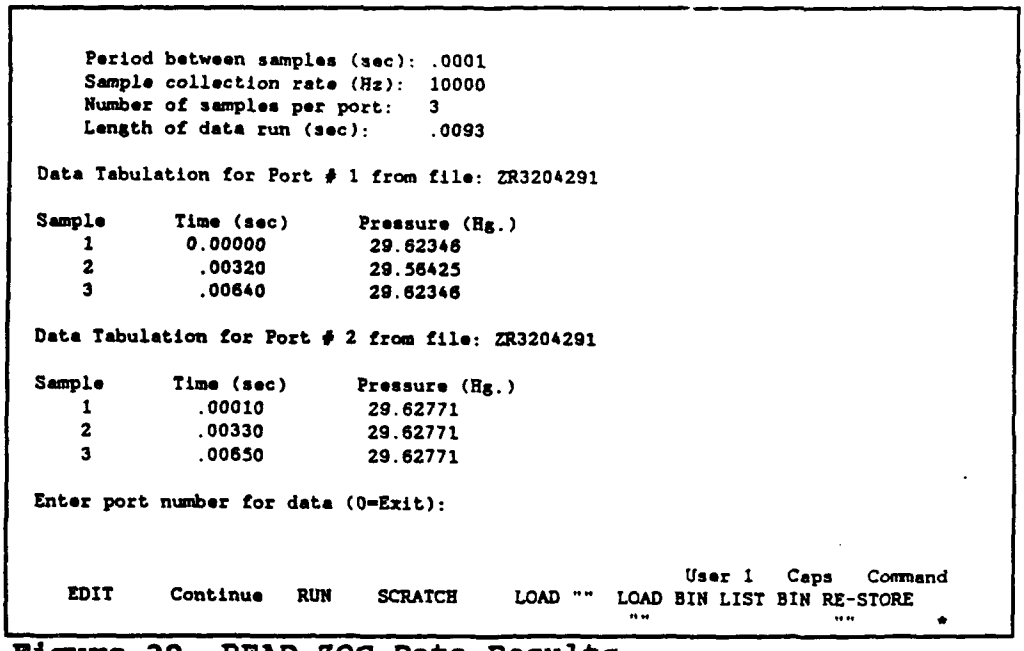


Figure 39 READ_ZOC Data Results

displayed pressure is the reduced pressure value calculated from the SCAN_ZOC_05 program. The output pressure is derived using the recorded transducer output voltage and the curve fitted to the calibration data for the identified port. The output values are seen to be within 1.3% of the applied pressure.

The program "PLOT_DATA" (Figure A19) is selected by function key F3 (PLOT DATA). The results from ZOC #3 are displayed in Figure 40 and Figure 41.

The program "CAL_READ_PR1" (Figure A20) is selected by function key F4 (READ CALSYS20). The results are displayed in Figure 42.

The program "TABULATE_ZOC" (Figure A21) is selected by function key F5 (TABULATE CAL DAT). This program displays the

```
Program plots reduced data from file ZR3204291

Statistics for Zoc # 3

    Period between samples (msec): .1
    Sample collection rate (Hz):  10000
    Number of samples per port:   3
    Length of data run (msec):     9.5


Data point can be plotted as a continuous line, or squares.

Note: Wait for symbol '*' in lower right corner of CRT to change
      to a '-' before pressing <Shift><Dump Graph>

                                                    User 1   Caps   Command
  EDIT     Continue   RUN    SCRATCH     LOAD ""  LOAD BIN LIST BIN RE-STORE
                                           ""                ""        *
```

Figure 40 PLOT_DATA Alpha Screen Display

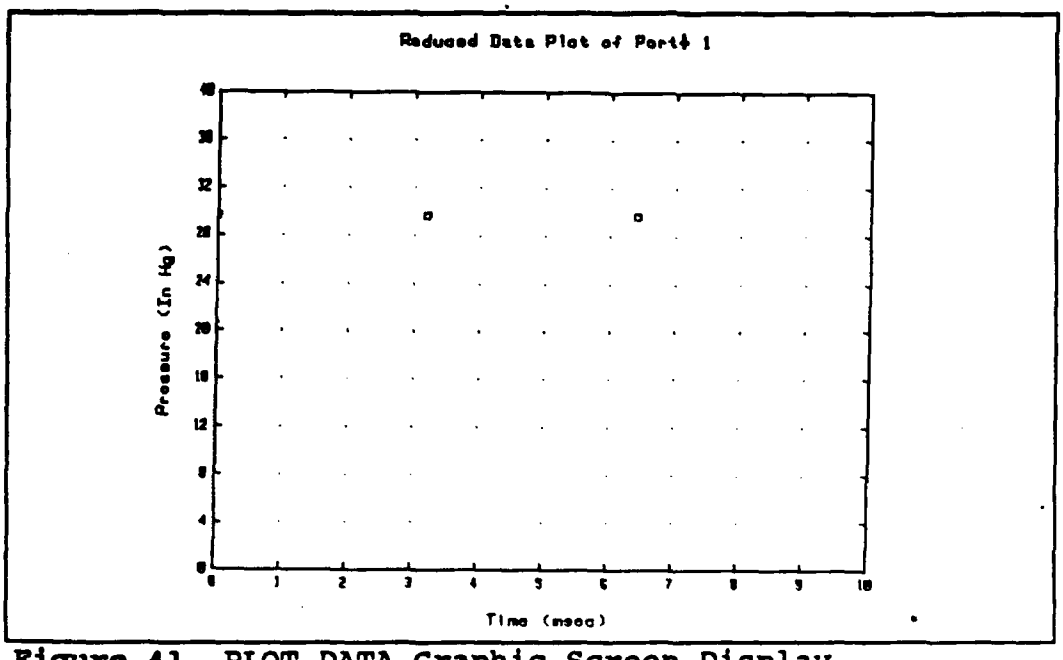


Figure 41 PLOT_DATA Graphic Screen Display

first three reduced data samples of selected ports, and the associated calibration data for each selected port. Figure 43 displays the results for ZOC #3.

```
Program: CAL_READ_PR1

   This program sequentially sets the CAL2000 calibration
   modes and reads the corresponding internal Pressure Standard
   for that mode.

         PH          Positive high range pressure to CAL(+)
         PM          Positive mid range pressure to CAL(+)
         PL          Positive low range pressure to CAL(+)
         ZO          CAL(+) & REF(-) connected together
         NL          Negative low range pressure to REF(-)
         NM          Negative mid range pressure to REF(-)
         NH          Negative high range pressure to REF(-)


CAL2000: Calibration modes and pressures.

  Mode     Pressure (in Hg)

   NH          30.1606
   NM          17.9945
   NL           7.3788
   ZO             .0034
   PL           7.3912
   PM          18.0194
   PH          30.2072
```

Figure 42 CAL_READ_PR1 Results

```
Program tabulates Zoc pressures and calibration data from
the SCAN_ZOC_05 program.
Reduced Data Tabulation at a sample rate of 10000  Hz

Port  Sample 1  Sample 2  Sample 3
   1    29.623    29.564    29.623
   2    29.628    29.628    29.628
   3     -.414     -.414     -.351
   4     -.495     -.425     -.425

Calibration Data Tabulation for Zoc# 3

Pressure voltage readings:
Port     NH       NM       NL       ZO       PL       PM       PH
  0 -30.154 -17.998  -7.382     .003    7.401   18.019   30.207
  1  -2.194  -1.310   -.521     .040     .723    1.688    2.704
  2  -2.584  -1.689   -.850    -.276     .450    1.478    2.549
  3  -1.853  -1.018   -.250     .272     .909    1.820    2.770
  4  -1.770  -1.036   -.362     .101     .673    1.485    2.320

Calibration polynomial coefficients for Zoc# 3
Port        A0              A1              A2              A3
 1  -1.00116186204  11.8955586742   -.435562698234   .106534210924
 2   2.45870376346  11.1095966035   -.349715337703   .0940454333066
 3  -3.96668335674  12.8254266665   -.523187753647   .119575469008
 4  -1.98202140047  14.2260022322   -.645728820628   .20262038018
```

Figure 43 TABULATE_ZOC Results

The program "LS_PLOT" (Figure A22) is selected by function key F6 (PLOT CAL DATA). Figure 44 displays the calibration data with voltage verses CALMOD calibration

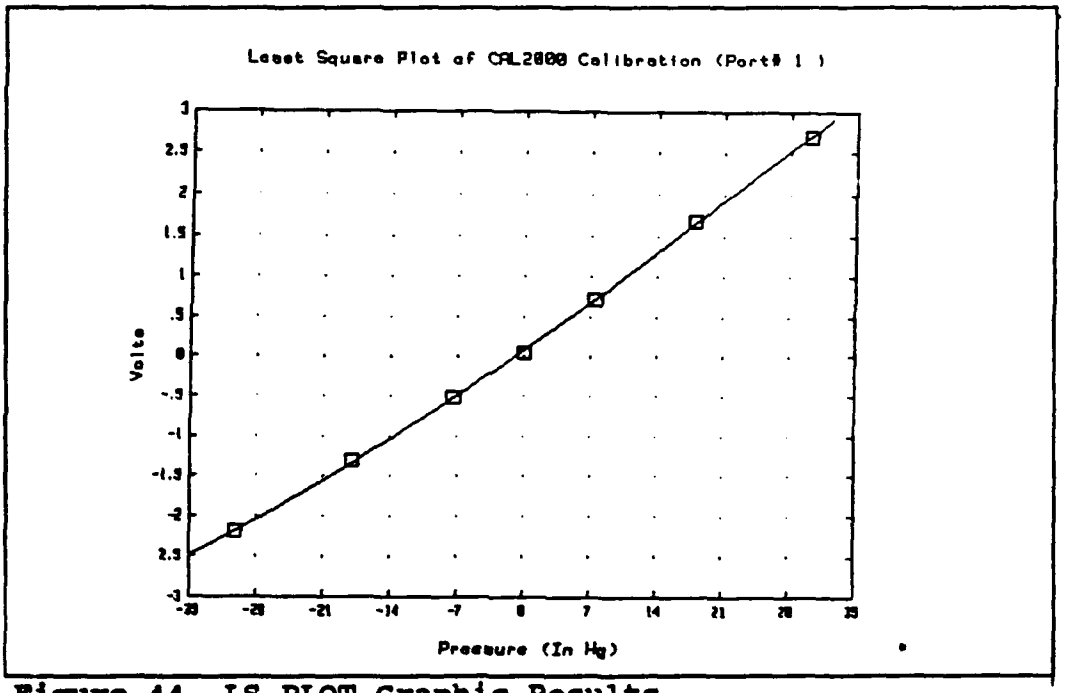


Figure 44 LS_PLOT Graphic Results

pressure (squares), and the curve fitted to the collected data by the least squares method (continuous line). The results are from port #1 on ZOC #3.

III. DISCUSSION

A. Upgraded Capabilities

The ZOC-14 DAS provides the capability to perform fully automatic high-speed pressure data aquisition, data storage and reduction. For the Gas Dynamics Laboratory (GDL), this upgraded capability allows very short run-times in the transonic and supersonic wind tunnels, resulting in greatly reduced electrical power consumption by the laboratories air compressors and the elimination of delays between wind tunnel tests. Additionally, wind tunnel models will have a longer useful life. The shorter run times will reduce model erosion and the net effect is a reduced cost to operate the wind tunnels and to maintain models.

The HP9000 now serves as the controller for the DAS's in each of the three laboratories comprising TPL and GDL. Each DAS includes Hewlett Packard HP-IB compatible instruments and the HG-78K Scanivalve Controller. Low speed pressure data acquisition, using the HG-78K to operate the Scanivalve rotary port pressure sensing unit, is retained. However, for turbomachinery measurements, phase-lock data acquisition using TPL's Digital Programmable Timing Device (or PACER), can not yet be performed using the HP9000 to replace the HP1000 [Ref. 24]. TPL's PACER is electrically hard-wired into the HP1000

as an I/O device. However, the integration of the HP9000 and HP6944A, as demonstrated in the ZOC-14 DAS, is known to provide the interface capability to operate the PACER.

Temperature and low-speed pressure measurement data acquisition and processing using the HP9000 has been demonstrated in turbocharger performance mapping laboratories and transonic wind tunnel thesis research [Ref. 25][16]. The HP9000 was shown to provide improved data storage handling and display capabilities (over the HP1000 system) using the HP9153 series hard/floppy disk drive and HP7475A plotter.

Development of the ZOC-14 DAS served to identify the programming and integration requirements, and the capabilities of the HP9000 and HP6944A as a subsystem. The HP9000/HP6944A as a controller and data acquisition sub-system provides to TPL and GDL the potential for extension in several prospective applications.

B. ZOC-14 DAS Outstanding Issues

The ZOC-14 DAS development is not fully complete. Several hardware, software, and performance issues need to be resolved, namely;

- Hardware: The CALSYS2000 is using only one CALMOD to provide calibration pressures for a specific ZOC operating range. A second CALMOD is required.

---

[16] Appendix D, Figures D3-D6 are HP9000 controlled Turbocharger Performance Mapping and data display programs.

- Hardware: It was not possible to set the "PURGE" mode to clear pressure sensing lines using nitrogen gas supply.
- Software: SCAN_ZOC_05 uses a "set" 1.5 sec time delay to account for calibration pressure stabilization during the calibration process, vice a technique to monitor the calibration pressure and digitize when found to be stable.
- Performance: The maximun data sampling rate acheivable with the ZOC modules needs to be verified.
- Performance: A data error analysis needs to be carried out.
- Performance: The calibration of the CALMOD Pressure Standard needs to be verified.

Some elaboration of each of these issues follows.

The use of the present one CALMOD results in only the 15 psid ZOC being calibrated over its full pressure range when 15 psid and 50 psid ZOCs are used in the DAS. The 50 psid ZOC must be used without the 15 psid ZOCs if they are to be calibrated and used over their full range. This would reduce the pressure data measurement availability from 96 (3x32) to 32 ports. Installation of a second CALMOD would allow independent calibration of the 15 psid and 50 psid ZOC modules. The program SCAN_ZOC_05 incorporates the steps to use two CALMODs, and requires no modifications.

The PURGE mode is set by providing control gas through both the Px and CAL control lines as illustrated in Figure 8[17]. Currently, the required control gas pressure is not provided by the PSC when the appropriate command is sent from

[17] Refer to Ref. 17, dwg 17750, PNEUMATIC DIAGRAM OF PSC.

the HP9000 to the CALMOD. The commands to set the PURGE mode are "aEC 8 Y" followed by "aEC 10 N" to set the solenoids in the PSC.

The calibration process requires a stabilization time delay to elapse between the setting of a calibration mode and the sampling of the calibration pressure by the Pressure Standard. The best method to determine the required time delay is to continuously sample the calibration pressure until the pressure stabilizes, then record the calibration pressure. The program SCAN_ZOC_05 uses an empirically derived 1.5 sec "wait period" between the calibration mode selections and the pressure sampling. This time value was determined using a modified CAL_READ_PR1 program, stopwatch and oscilloscope. CAL_READ_PR1 was modified with a PAUSE statement to allow observation of the ZOC voltage signal on the oscilloscope. When the signal appeared to stabilize on the oscilloscope, the CONTINUE key (F2) was depressed to sample the pressure and select the next mode. The times for six complete mode selections were recorded. The average time was found to be 1.34 seconds between depressions of the CONTINUE key. The value of 1.5 seconds was selected as a conservative value to use for the time delay.

The maximum sample data collection rate has not yet been realized with the ZOC-14 DAS. Preliminary observations revealed a random fall-off in pressure values when the "input" data sampling rate was greater than 50 KHz. A more careful

analysis of the sampling rate is required to validate the current 50 KHz value and to determine the maximum data sampling rate.

An error analysis of the ZOC-14 DAS is required to determine the accuracy of the data collection and reduction process. The HP67959A A/D Cards have a documented resolution of five milli-volts for their factory set configuration as installed in the HP6944A. The Pressure Standard mounted in the CALMOD has accuracy specifications documented in Reference 17. The ZOCs have measurement accuracy specifications documented in Reference 16. The numerical Least-Squares curve fitting routine introduces a so-far unspecified uncertainty. The reduced pressure readings provided in Figure 39 compared to the 30.0 inches of mercury source pressure, provide only a single example of the pressure measurement uncertainty given by the DAS[18]. Derivation of the overall system uncertainty needs careful attention.

The Pressure Standard's calibration has not been verified since it's delivery. Calibration verification is essential prior to accepting pressure measurement data. Accordingly, the CALMOD calibration coefficients, derived from the calibration verification, need to be changed on the CALMOD

---

[18] Uncertainty in the source pressure of 30.0 inches of mercury as measured by the TPL Calibration Pressure Manometer needs to be considered in the uncertainty analysis.

EPROM. Reference 17 provides the details to conduct Pressure Standard calibration and coefficient changes on the EPROM.

C. Potential Extensions and other Applications

The HP9000/HP6944A subsystem provides the hardware features necessary to integrate computer control of experiments with the data acquisition process. Three immediate applications using the HP9000/HP6944A are identified here:

- The TPL PACER can be interfaced with the HP9000/HP6944A to provide phase-locked data acquisition capability[19].
- Kulite pressure probe measurements can be acquired using the analog signal from the conditioning amplifier. The signal would be routed directly to the Buffer A/D Function input connector, using the Timer Function to set the data collection rate and number of samples.
- GDL's transonic and supersonic wind tunnels can be operated with fully automated data acquisition and experiment control systems. The tunnel's manual back-pressure valve, and electro-servo controlled translating survey probe can be operated by the HP9000/HP6944A to give fully automated control.

---

[19] Preliminary TPL PACER interfacing techniques have been examined but not yet documented.

IV. CONCLUSIONS

The DAS upgrade and extension which is reported here has included the adoption of the HP9000 as a controller for existing HP-IB compatible DAS instrumentation, the generation of acquisition and reduction software for the existing system, and the development of a new high-speed pressure data acquisition capability. The new high-speed system involved an integration of the HP9000 with the HP6944A Multiprogrammer and with Scanivalves's ZOC-14 and CALSYS2000 systems. In the present account, emphasis has been placed on reporting the development of the ZOC-14 DAS. The hardware and software for the system have been successfully demonstrated. It has been shown that the use of the system in the Gas Dynamics Laboratory can reduce wind tunnel test times by a factor of 20.

Six issues concerning the present hardware and software have been identified as needing to be resolved, and immediate applications of the HP9000/HP6944A system to Kulite and Phase-Locked data acquisition, probe survey and tunnel condition control, have been identified. The programs and experience reported in the present document can serve to guide these extensions.

The key to developing the capabilities resident in the HP9000/HP6944A system is a thorough familiarity with HP BASIC

and HP14753A CAT programming. Therefore, the recommendation is made that formal instruction in this language and programming techniques be provided before the recommended extensions of the system are attempted.

APPENDIX A. ZOC-14 PROGRAMS

Appendix A is a compilation of pertinent information and programs used to operate the ZOC-14 DAS.

Figure A1 is the ZOC-14 DAS program configuration file, "ZOC_CONFIG_05", print-out for the I/O Cards installed in the HP6944A. The print-out is produced using the CAT program "DOCUMENT". The DOCUMENT program is located on the "/HP6944A" directory (Figure D1). ZOC_CONFIG_05 is a BDAT file located in the /HP6944A directory.

Figures A2 through A16 are parts of the SCAN_ZOC_05 program flow chart. Figure A17 is the SCAN_ZOC_05 program listing with program-specific remarks anotated after the "!" character.

Figures A18 through A22 are ZOC-14 DAS utility programs located on the /HP6944A directory.

Figure A23 is the program used to display the ZOC operation menu, and define the function keys to provide menu item selection by function keys.

```
File Name: ZOC_CONFIG_05
User ID:                          26 Apr 1992 16:53:12
List of Names:
  Buffer1          Adc1             Buffer2
  Adc2             Buffer3          Adc3
  Timer
```

```
**************************************************
        Configuration for Buffer1
        Model Buffer
**************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 0
Initial Mode ..... FIFO
Initial Lockout ..... Off
Timeout ..... 10
Memory Type .... 69791A
No of Extenders ... 0
Ref Reg 1 .. 0
Ref Reg 2 ... 0
Buffer Direction .. In
Front End Type .. 69759A
A/D Names:   1 . Adc1
2 . None
3 . None
4 . None
5 . None
6 . None
7 . None
8 . None

**************************************************
        Configuration for Adc1
        Model 69759A 500 KHz A/D
**************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 7
Full-scale Range ..... +-10.24 volts
Initial Internal Range ... 10
Range Source ..... Internal
Scanner ..... None
Timeout ..... 10
Data Conversion ..... Standard
Internal Trigger ..... Disabled
External Trigger ..... Enabled
Trigger Mode ..... Multiple
Trigger Polarity ..... Negative
Lockout Polarity ..... Disabled
Master Output Enable ..... Enabled
Mux Output Control ..... Disabled
External Output Enable .... Disabled
Gate Mode ..... Multiple
Return Data With ..... Sign Extension
```


Figure A1 ZOC-14 Configuration File

```
*****************************************************
        Configuration for Buffer2
        Model Buffer
*****************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 2
Initial Mode ..... FIFO
Initial Lockout ..... Off
Timeout ..... 10
Memory Type .... 69791A
No of Extenders ... 0
Ref Reg 1 .. 0
Ref Reg 2 ... 0
Buffer Direction .. In
Front End Type .. 69759A
A/D Names:   1 . Adc2
2 . None
3 . None
4 . None
5 . None
6 . None
7 . None
8 . None

*****************************************************
        Configuration for Adc2
        Model 69759A 500 KHz A/D
*****************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 9
Full-scale Range ..... +-10.24 volts
Initial Internal Range ... 10
Range Source ..... Internal
Scanner ..... None
Timeout ..... 10
Data Conversion ..... Standard
Internal Trigger ..... Disabled
External Trigger ..... Enabled
Trigger Mode ..... Multiple
Trigger Polarity ..... Negative
Lockout Polarity ..... Disabled
Master Output Enable ..... Enabled
Mux Output Control ..... Disabled
External Output Enable .... Disabled
Gate Mode..... Multiple
Return Data With ..... Sign Extension
```


Figure A1 (cont) ZOC-14 Configuration File

```
**************************************************
        Configuration for Buffer3
        Model Buffer
**************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 4
Initial Mode ..... FIFO
Initial Lockout ..... Off
Timeout ..... 10
Memory Type .... 69791A
No of Extenders ... 0
Ref Reg 1 .. 0
Ref Reg 2 ... 0
Buffer Direction .. In
Front End Type .. 69759A
A/D Names:   1 . Adc3
2 . None
3 . None
4 . None
5 . None
6 . None
7 . None
8 . None

**************************************************
        Configuration for Adc3
        Model 69759A 500 KHz A/D
**************************************************
Multiprogrammer Type ... HP6954A or HP6944A
Interface Select Code .. 29
Frame ..... 0
Slot ..... 11
Full-scale Range ..... +-10.24 volts
Initial Internal Range ... 10
Range Source ..... Internal
Scanner ..... None
Timeout ..... 10
Data Conversion ..... Standard
Internal Trigger ..... Disabled
External Trigger ..... Enabled
Trigger Mode ..... Multiple
Trigger Polarity ..... Negative
Lockout Polarity ..... Disabled
Master Output Enable ..... Enabled
Mux Output Control ..... Disabled
External Output Enable .... Disabled
Gate Mode..... Multiple
Return Data With ..... Sign Extension
```


Figure A1 (cont) ZOC-14 Configuration File

```
*****************************************************
        Configuration for Timer
        Model Timer
*****************************************************
Multiprogrammer Type ... HP6934A or HP6944A
Interface Select Code .. 29
Timer-Pacer Frame .. 0
Timer-Pacer Slot .. 13
Counter Frame ..... 0
Counter Slot ..... 15
Counter size ... 15 Bits
Timeout ........ 10
Data Conversion ........... Standard
Run Time Limit Checking ... No
Initial Period ........ 2.E-6
Initial Count ......... 0
Digital Out Controller .. No
```


Figure A1 (cont) ZOC-14 Configuration File

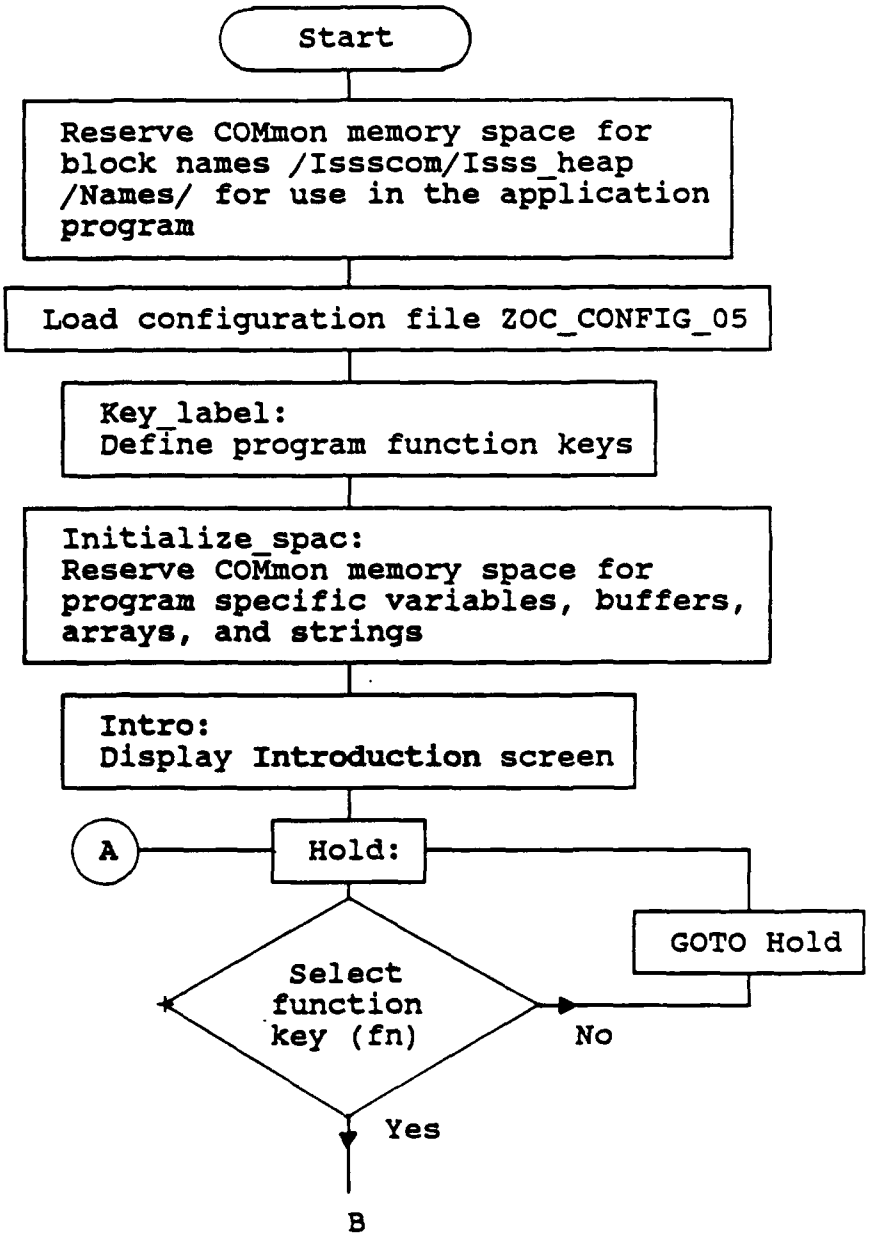


Figure A2 Program: Start-up and Initialization

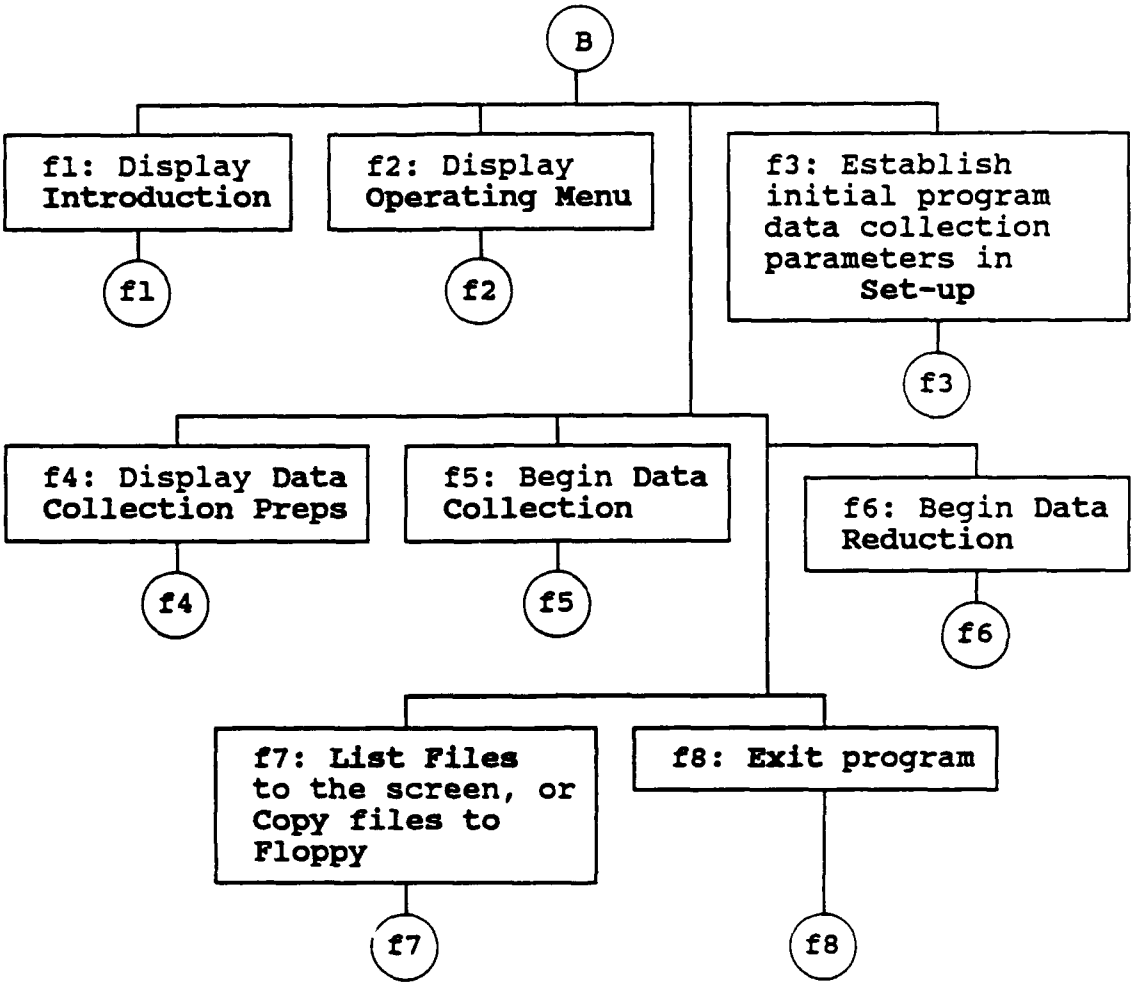


Figure A2 (cont) Program: Start-up and Initialization

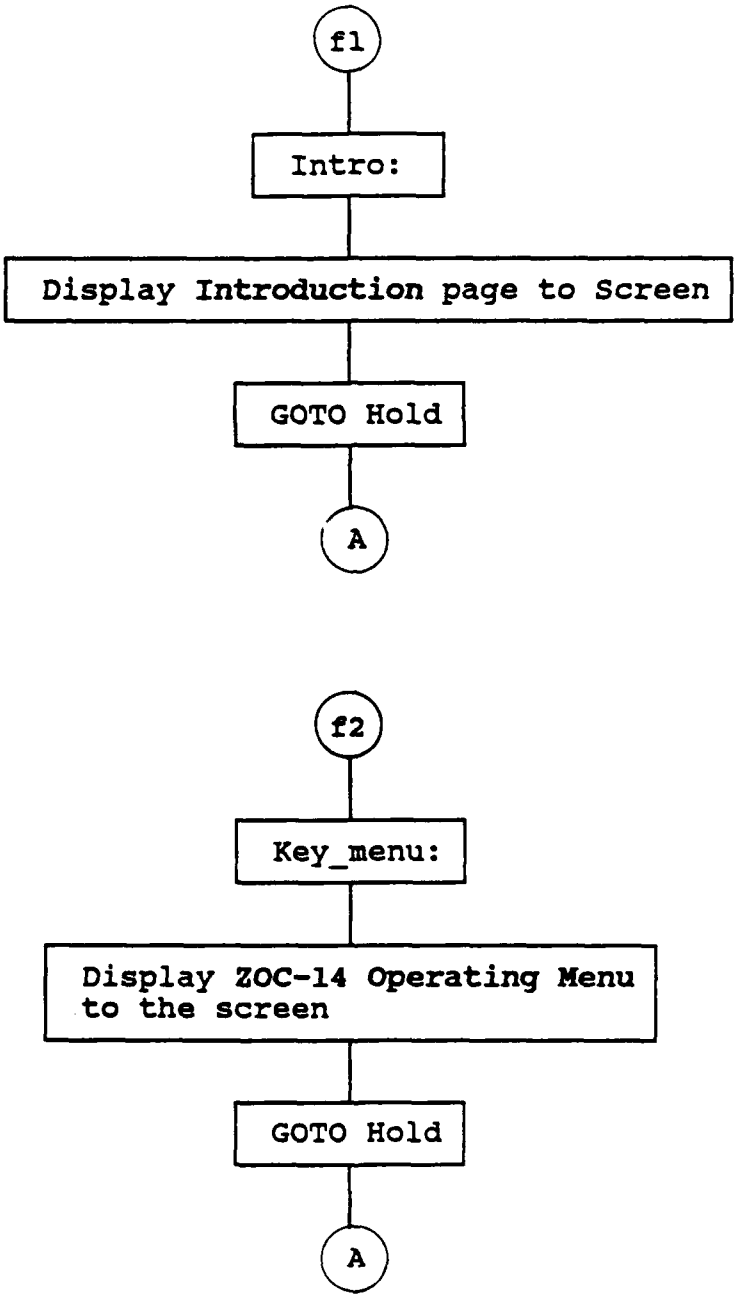


Figure A3 Program: Introduction and Operating Menu

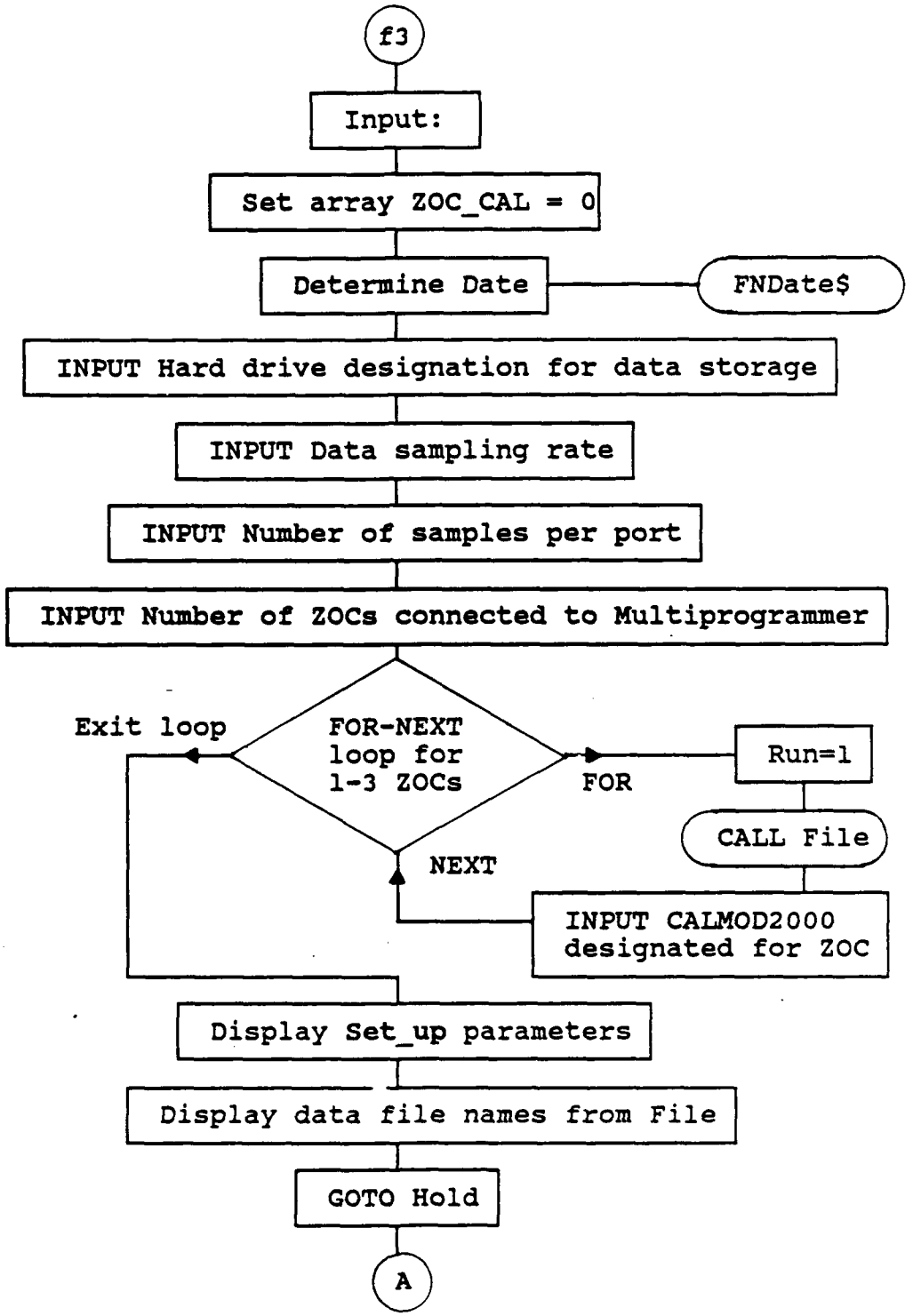


Figure A4 Program: Set-up Parameters

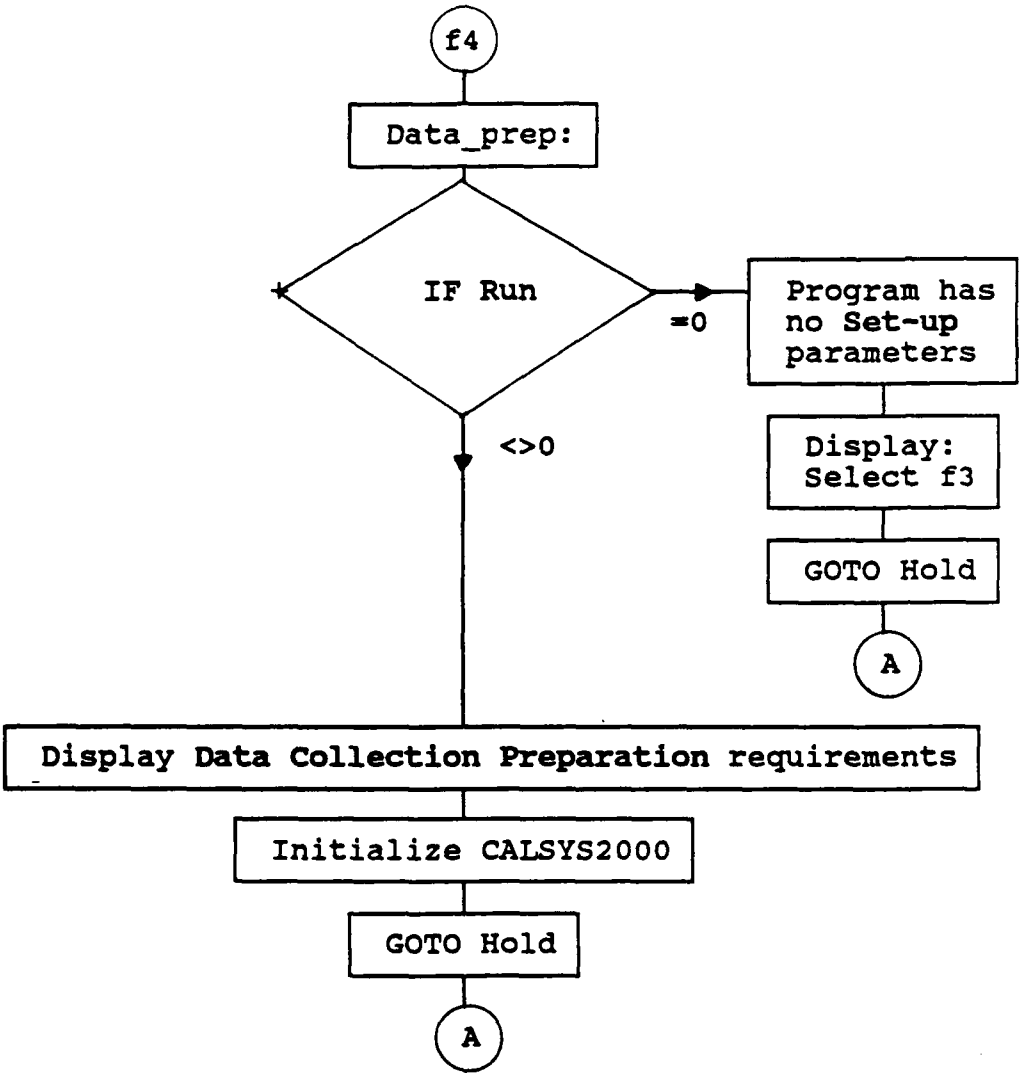


Figure A5 Program: Data Collection Preparations

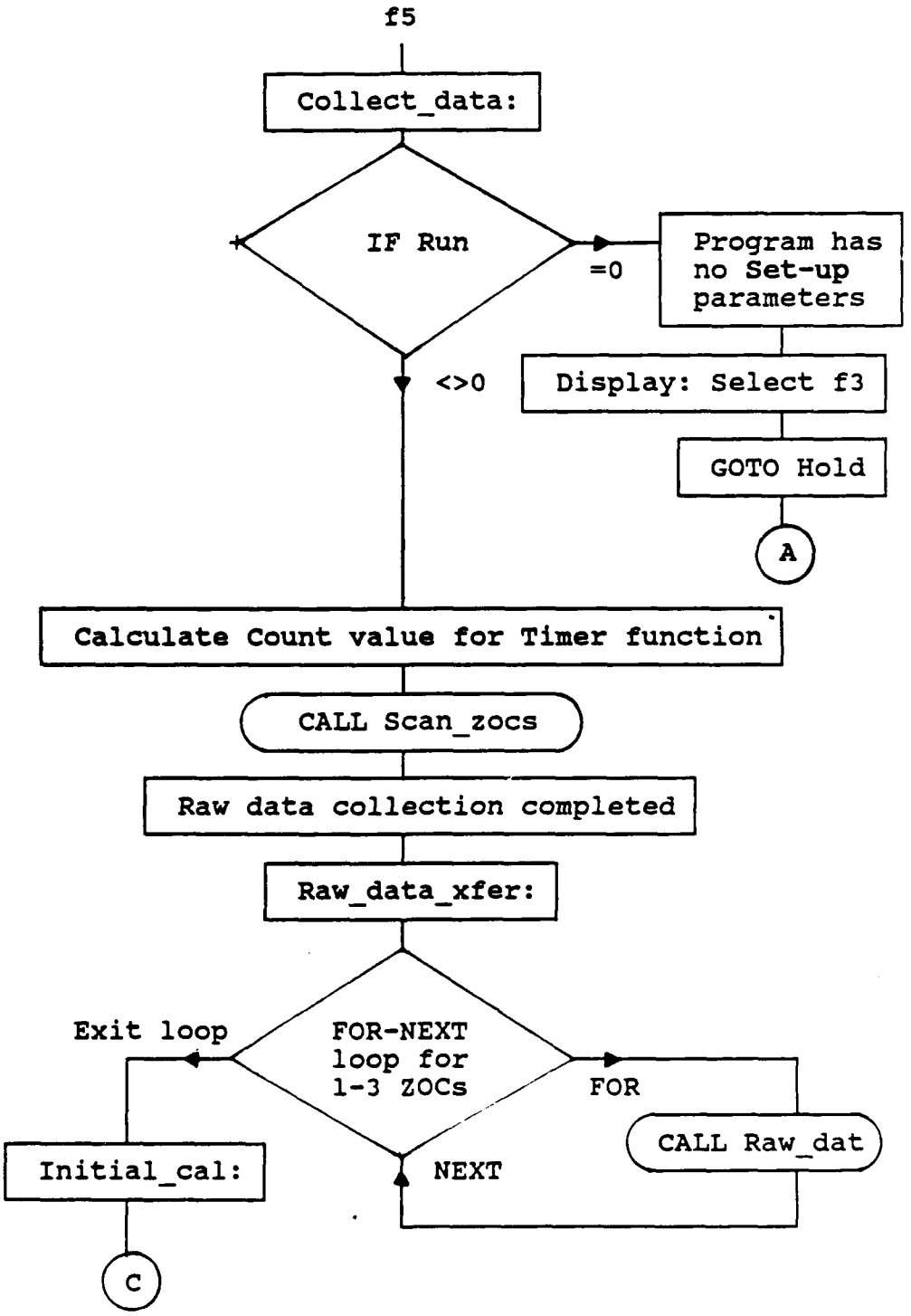


Figure A6 Program: Data Collection

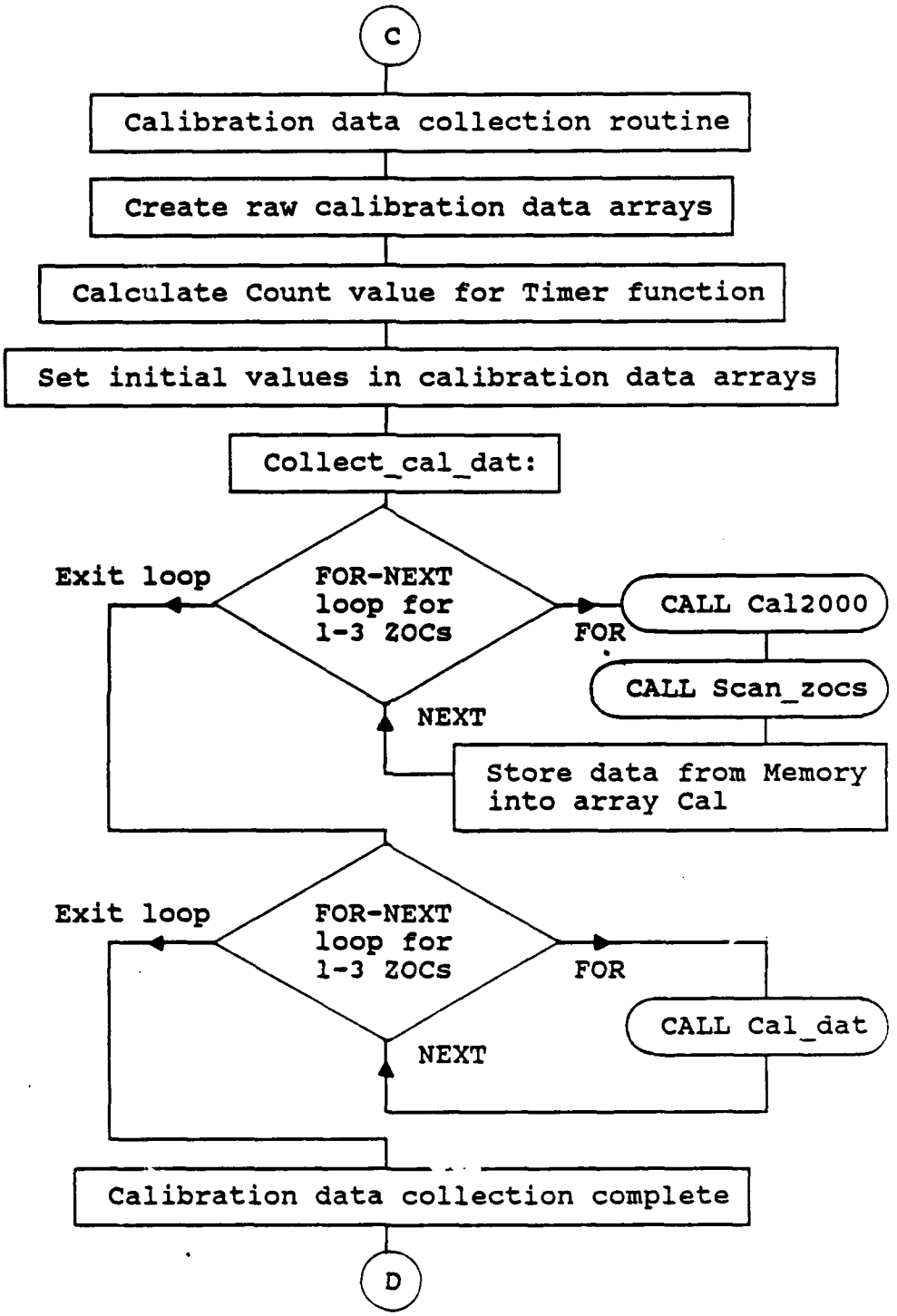


Figure A6 (cont) Program: Data Collection

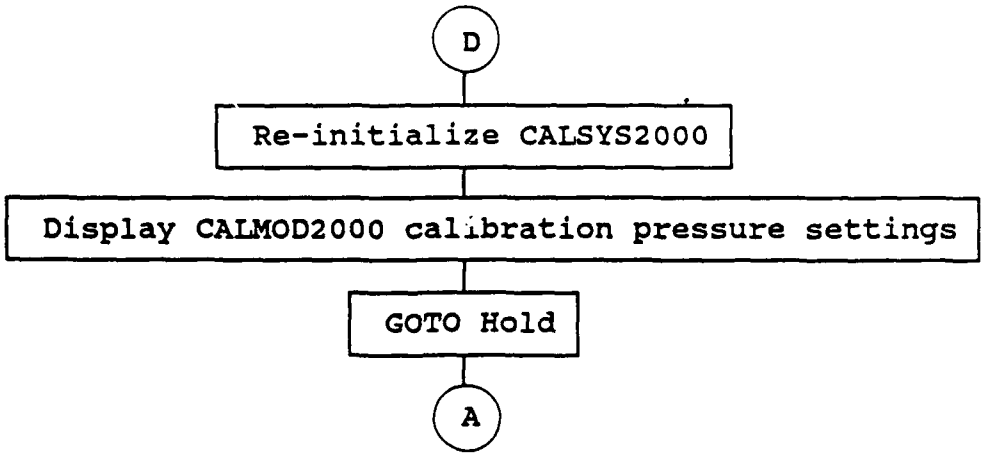


Figure A6 (cont) Program: Data Collection

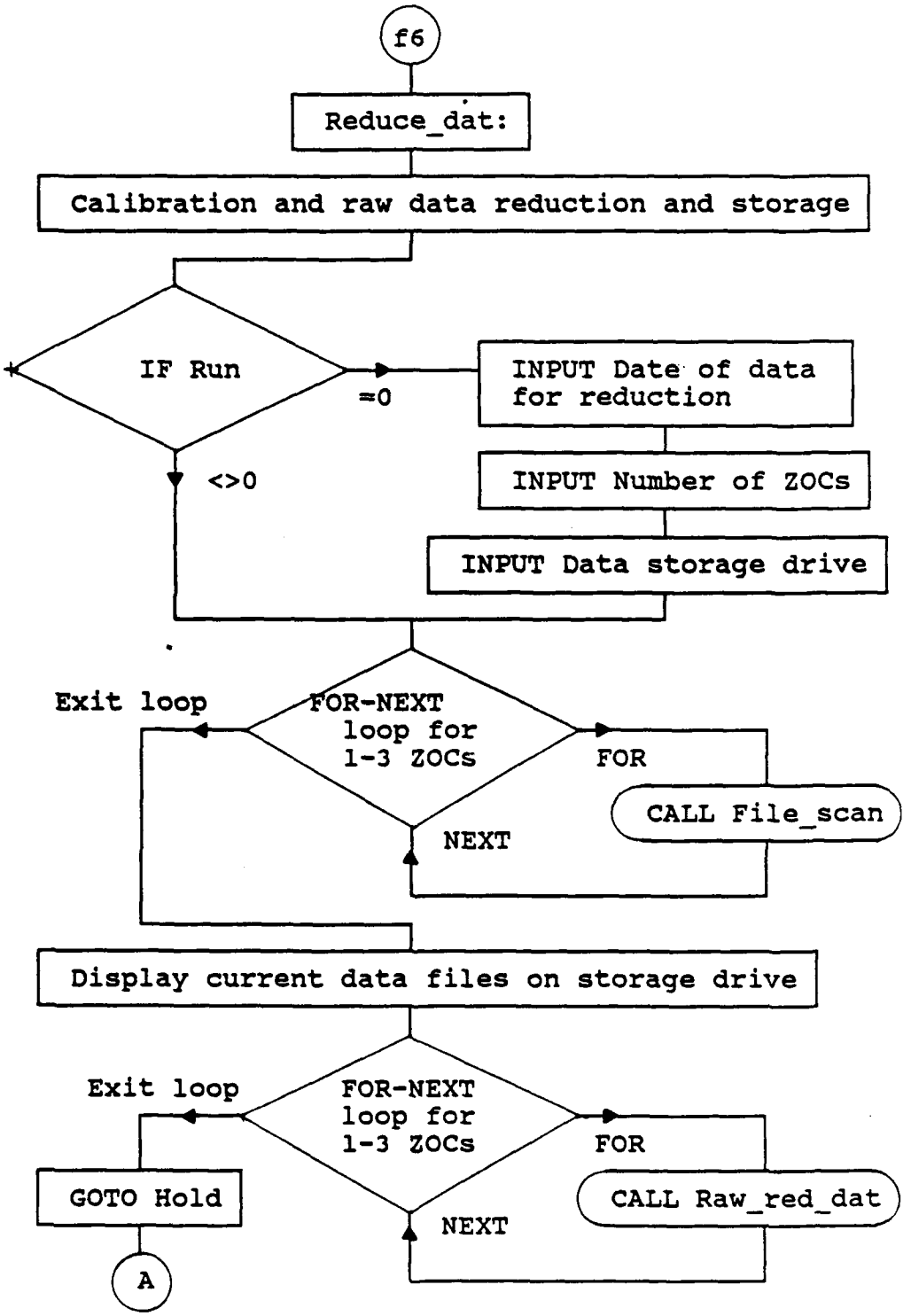


Figure A7 Program: Data Reduction and Storage

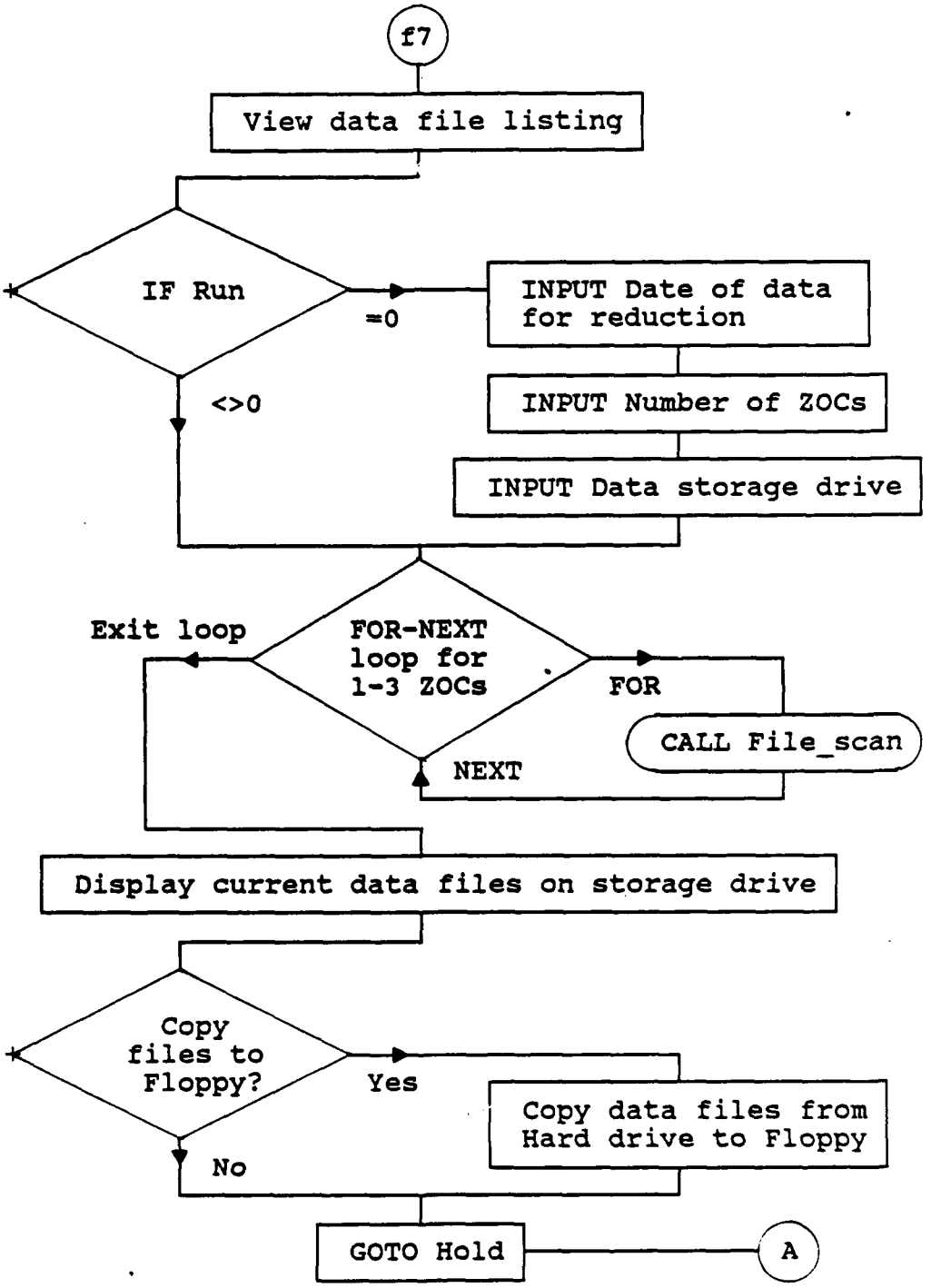


Figure A8 Program: Data File Listing and Storage

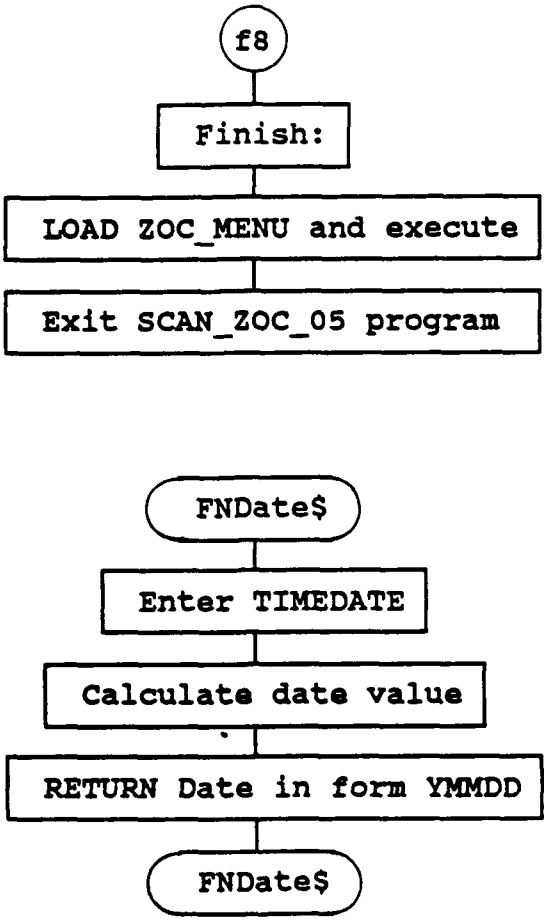


Figure A9 Program: Exit / Subprogram: FNDate$

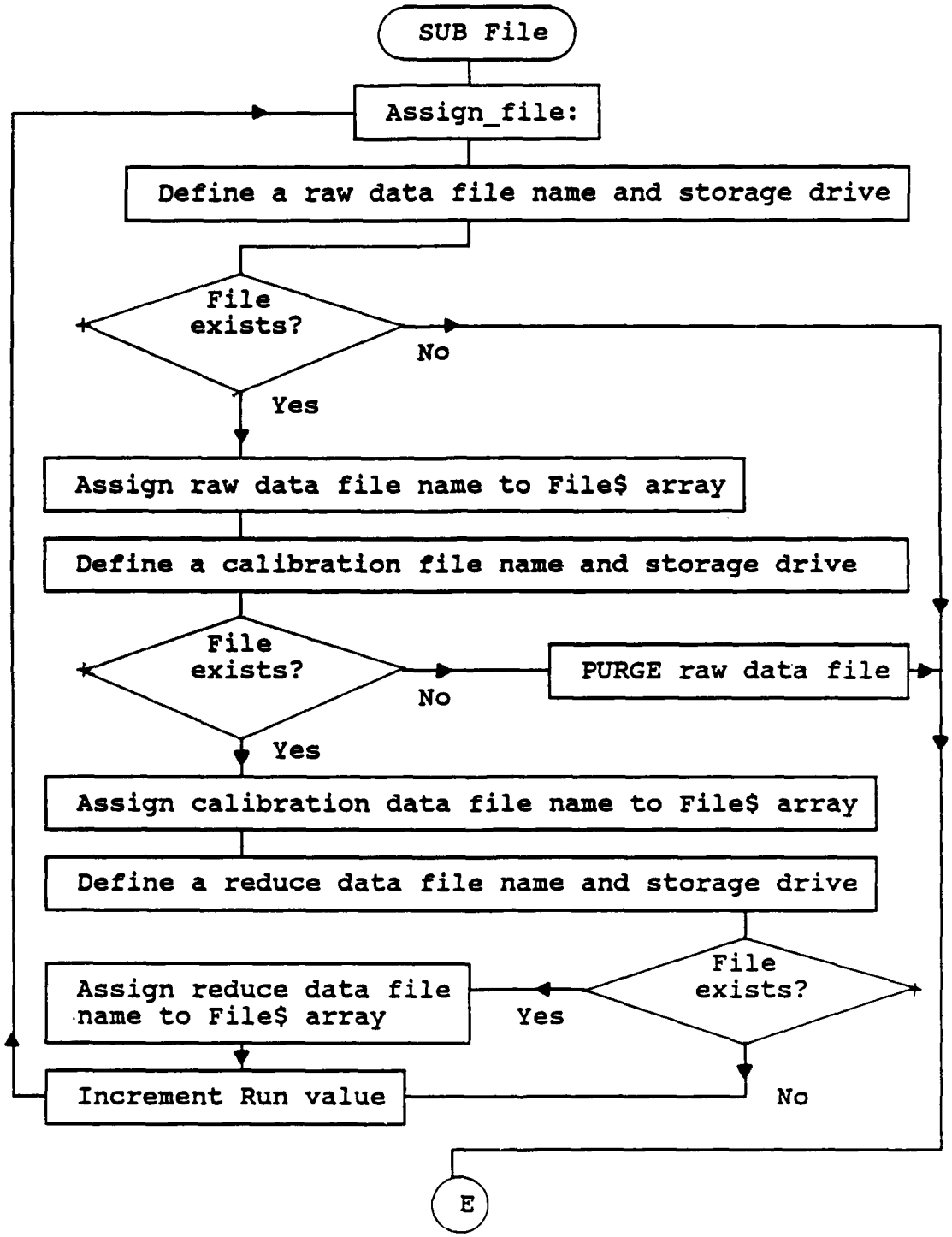


Figure A10 Subprogram: File

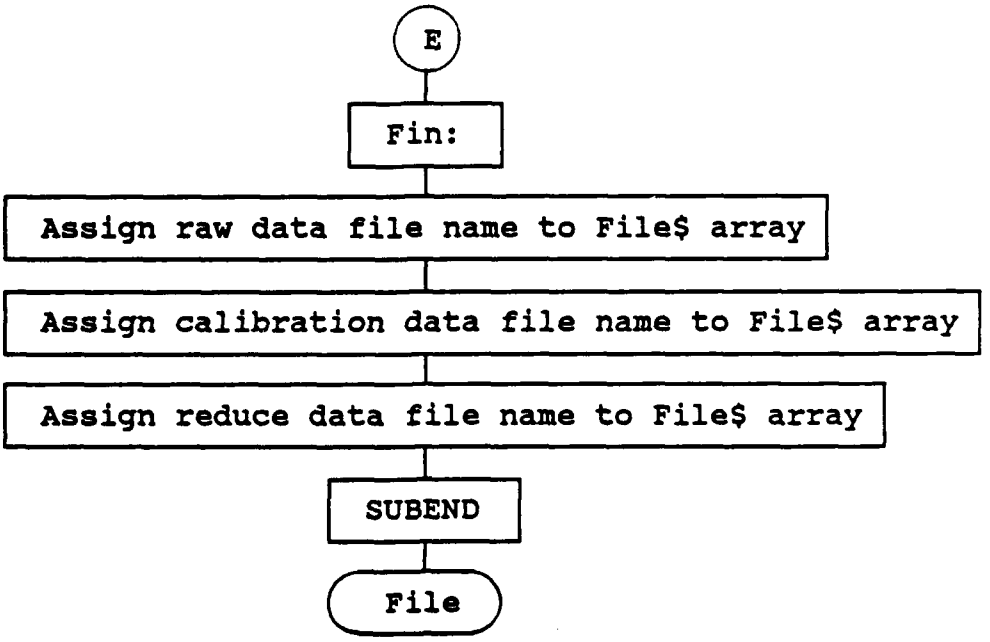


Figure A10 (cont) Subprogram: File

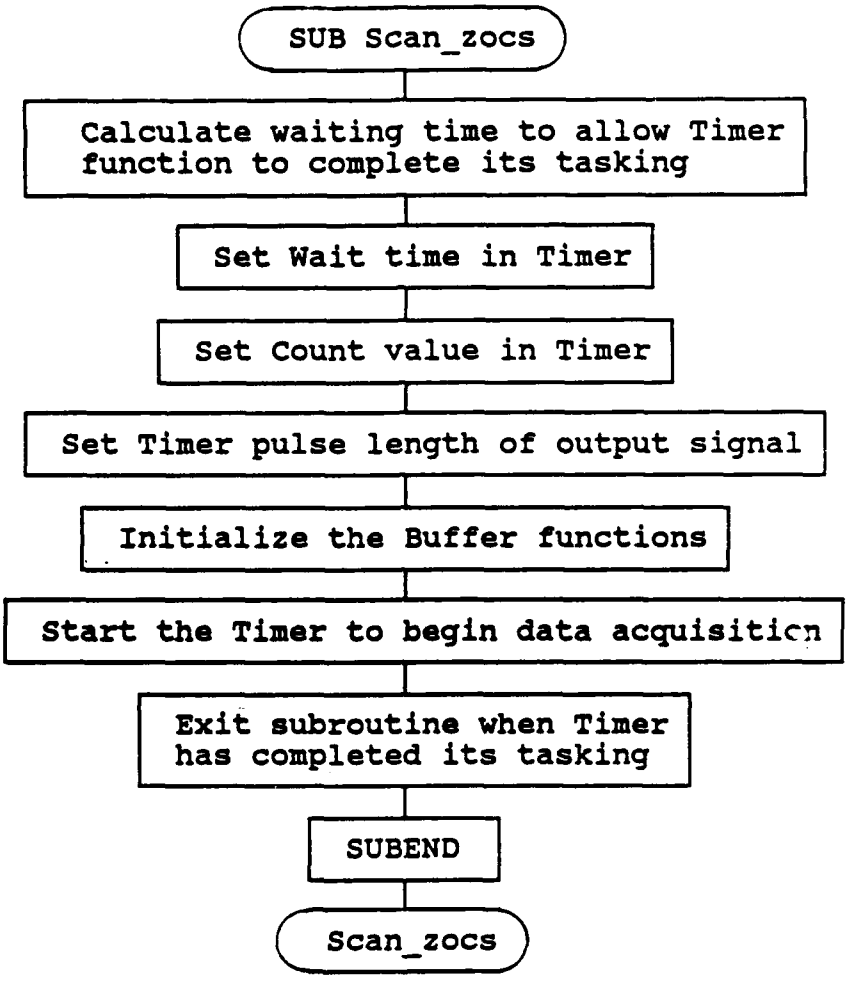


Figure A11 Subprogram: Scan_zocs

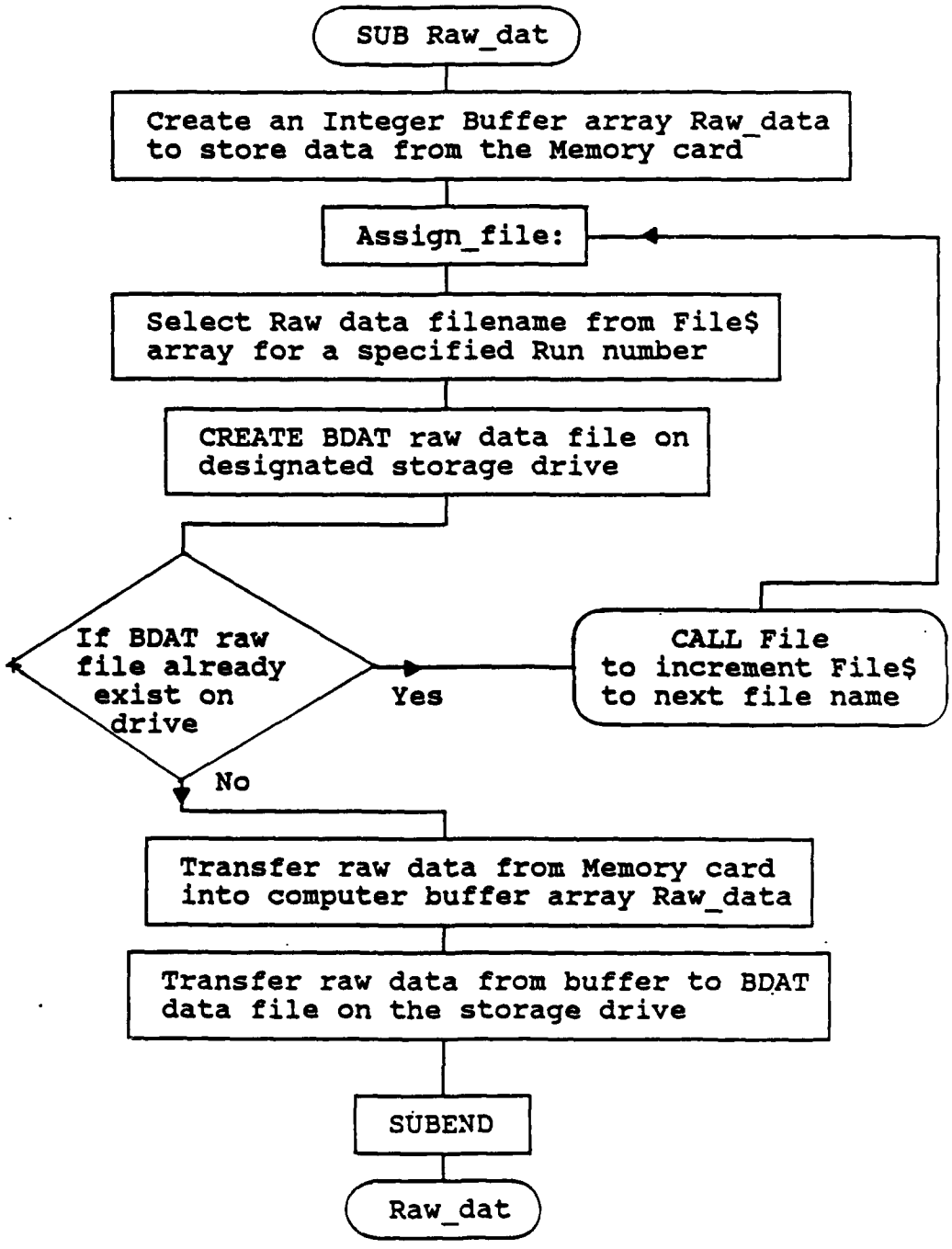


Figure A12 Subprogram: Raw_dat

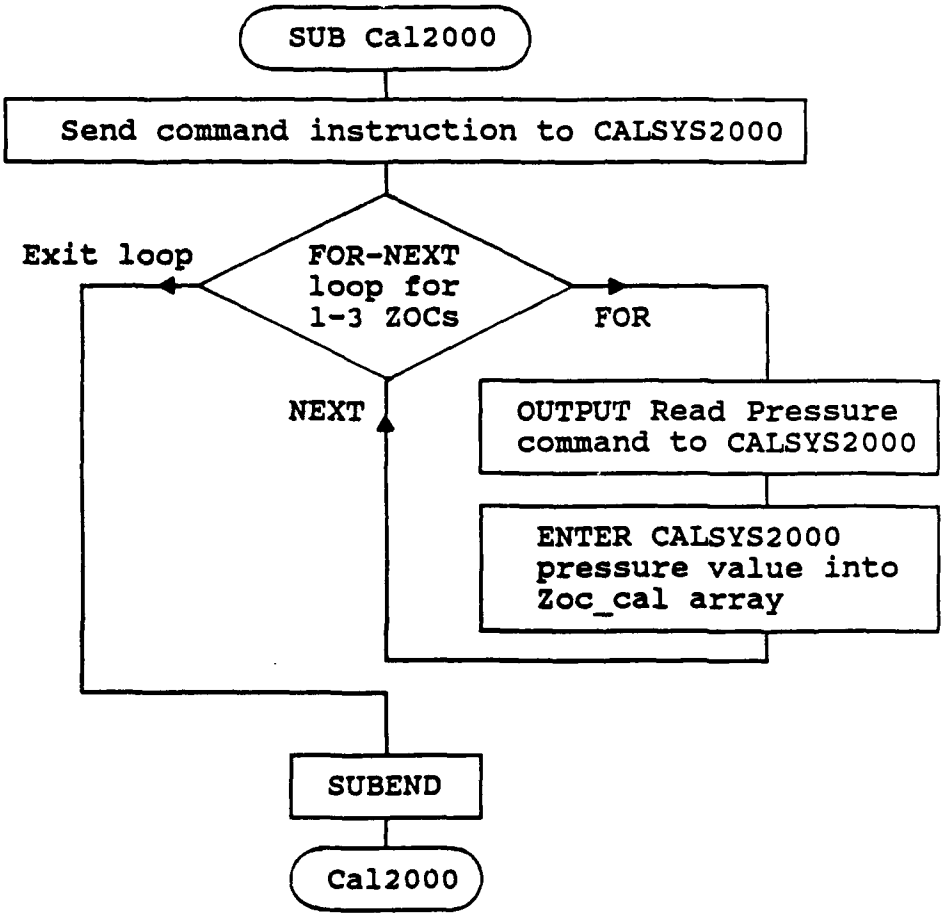


Figure A13 Subprogram: Cal2000

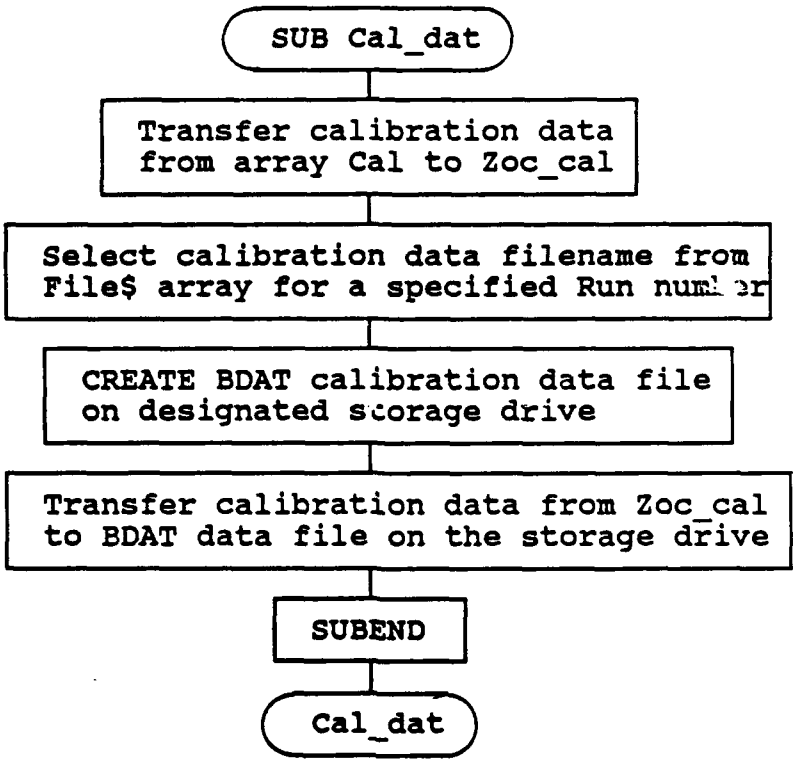


Figure A14 Subprogram: Cal_dat

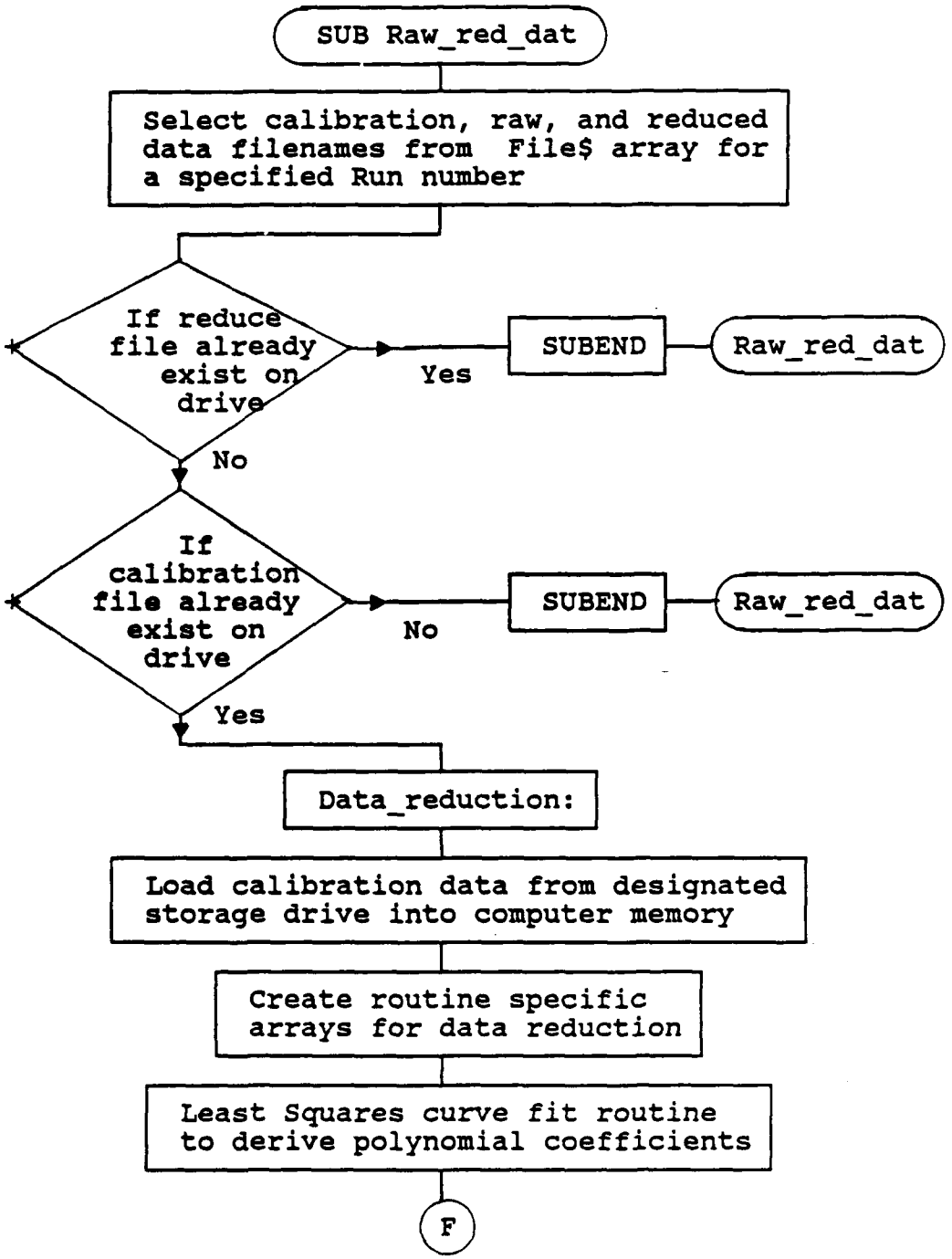


Figure A15 Subprogram: Raw_red_dat

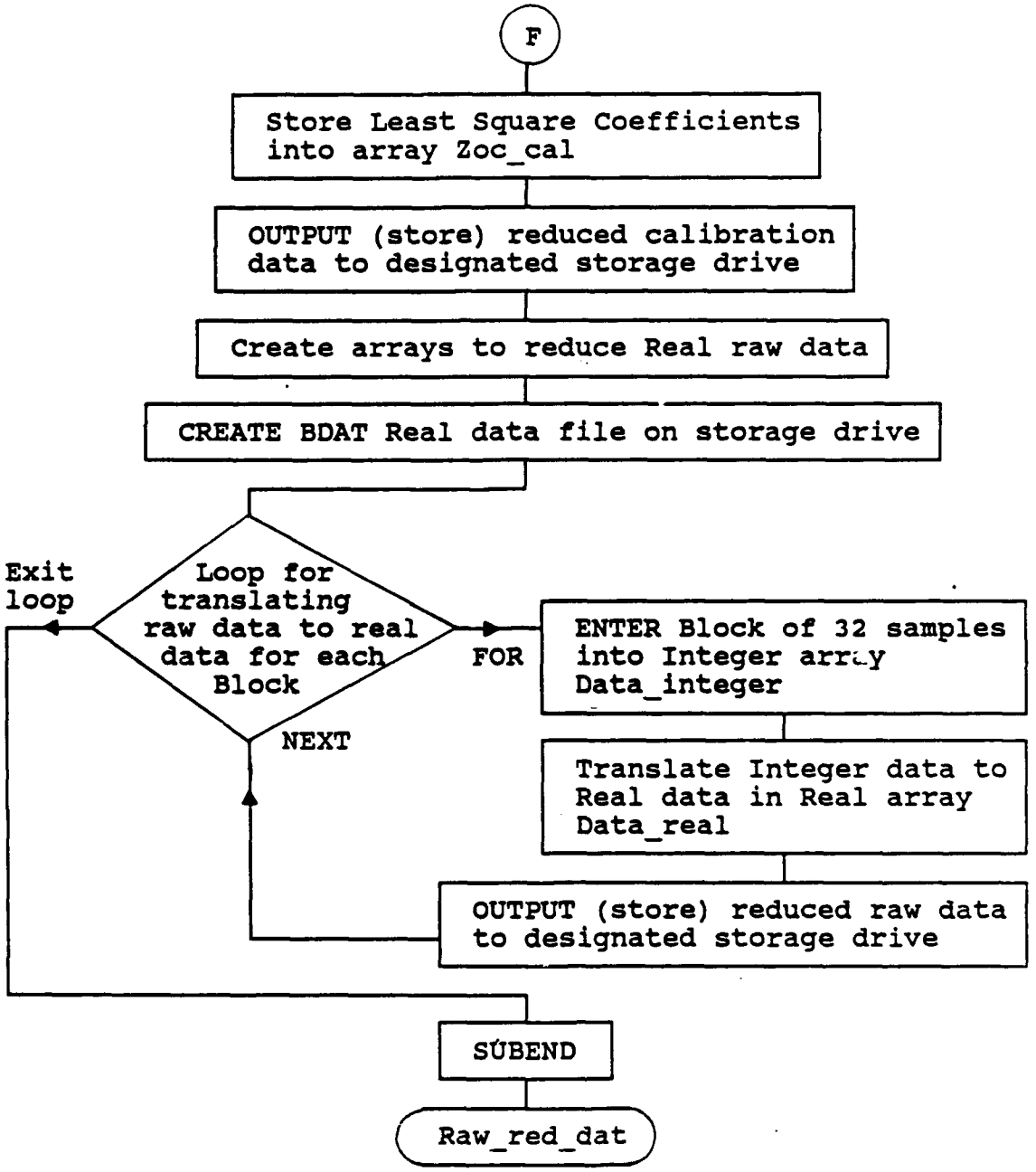


Figure A15 (cont) Subprogram: Raw_red_dat

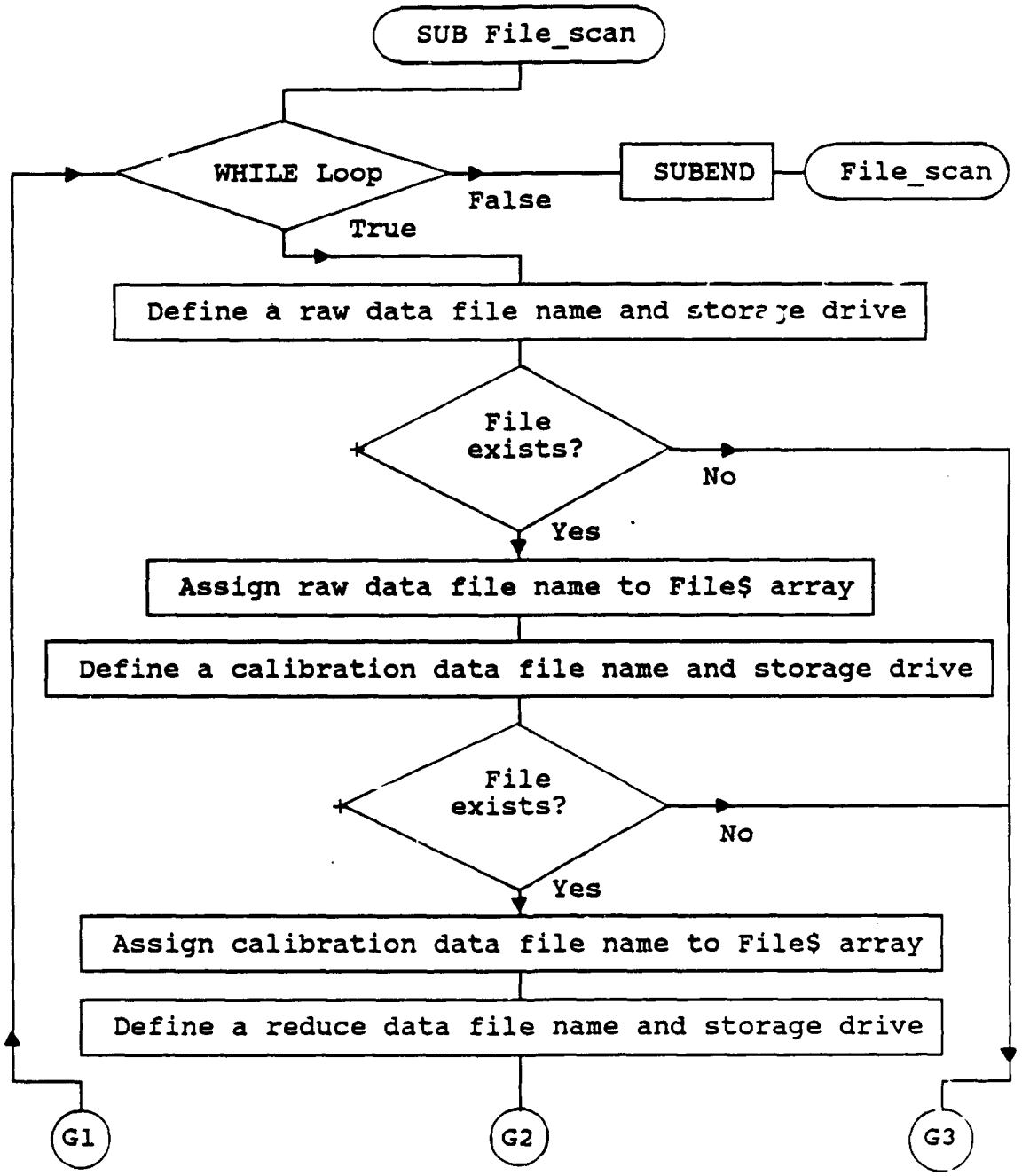


Figure A16 Subprogram: File_scan

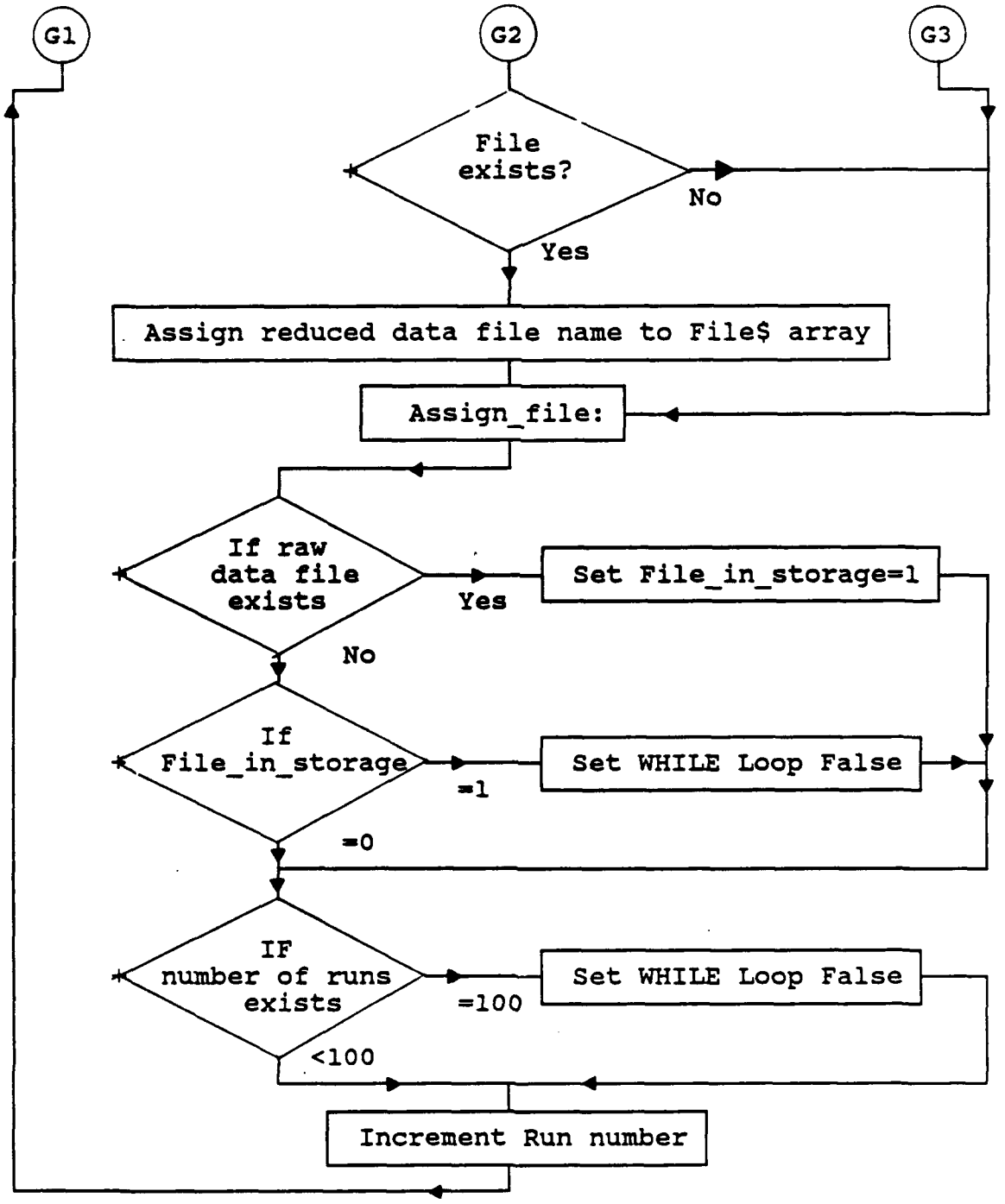


Figure A16 (cont) Subprogram: File_scan

```
10   ! Program: SCAN_ZOC_05
20   ! Description: Application program to operate HP6944A collecting pressure
30   !              readings from 1-3 ZOC-14 32 port modules using the CALSYS
40   !              2000 to provide calibration data, reduce raw pressure data
50   !              and store data to the hard drive.
60   ! Hardware:  (1) HP6944A Multi-processors
70   !             - (3) 500 kHz A/D Cards (HP69759A)
80   !             - (3) High Speed Memory Cards (HP69791A)
90   !             - (1) Timer/Pacer Card (HP69736A)
100  !             - (1) Counter Card (HP69775A)
110  !            (1) HiScan CALSYS 2000 Calibration Module
120  !            (3) ZOC-14 32 port Electronic Pressure Scanning Modules
130  ! Notes: 1. This program utilizes up to three (3) ZOC Modules storing data
140  !        of each ZOC into a seperate buffer Memory System (HP69791A).
150  !        2. COM /Names/ line and BDAT file ZOC_CONFIG_05 must match for
160  !        this program to operate.
170  !        3. CALSYS2000 requires a short period to stabilize before reading
180  !        the pressure valves.  The Pause_for statement sets (line 470) this
190  !        wait period in seconds.  Adjustment of the variable my be required
200  !        as additional ZOCs are integrated into the Data Acquisition System.
210  !        4. CALSYS2000 currently configured for one (1) calibrator.  This
220  !        program is written to operate one (1) or two (2) calibrators.
230  !
240  ! Buffer Memory: 65536 16-bit data words in HP69791A per system
250  ! Timer: Maximum 32767 counts for one HP69775A (32*1023=32736 data points)
260  ! Max speed of HP system is Period=0.000002 sec. or 500 kHz.
270  !
280   COM /Isascom/ INTEGER X(1:1106)
290   COM /Isas_heap/ Isas_heap(1000)
300   COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Buffer3,Adc3,Timer
310   Configure("Menu_off","ZOC_CONFIG_05")
320  !Configure("Ask_me","ZOC_CONFIG_05")
330   !
340 Key_label: !------ KEY LABEL ASSIGNMENT -----------------------------------
350           !
360   KEY LABELS ON
370   ON KEY 1 LABEL "Intro" GOTO Intro
380   ON KEY 2 LABEL "Key     Menu" GOTO Key_menu
390   ON KEY 3 LABEL "Set-up" GOTO Input
400   ON KEY 4 LABEL "Data    Preps" GOTO Data_prep
410   ON KEY 5 LABEL "Collect Data" GOTO Collect_data
420   ON KEY 6 LABEL "Reduce  Data" GOTO Reduce_data
430   ON KEY 7 LABEL "List    Copy" GOTO View_files
440   ON KEY 8 LABEL "Exit" GOTO Finish
450   !
460 Initialize_spac:  !----- ASSIGN MEMORY SPACE ------------------------------
470   Pause_for=1.5                       ! Wait time for CALSYS2000 stabilization
480  ! COM assigns calibration data array for 32 Zoc ports and standard values.
490   COM /Zoc_dat/ REAL Zoc_cal1(33,10) BUFFER,Zoc_cal2(33,10) BUFFER,Zoc_cal3(33,10) BUFFER
500   COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),DateS[6],Run
510   COM /Files/ FileS(1:99,1:9)[14],Data_driveS[11] !Data file & storage drive.
520  !
530   DIM Command_modeS(1:7)[2]
540   Command_modeS(1)="NH"
550   Command_modeS(2)="NM"
560   Command_modeS(3)="NL"
570   Command_modeS(4)="ZO"
580   Command_modeS(5)="PL"
590   Command_modeS(6)="PM"
600   Command_modeS(7)="PH"
610   !
620   Run=0
630   Data_reduced=0
640   !
650 Intro: !----- INTRODUCTION SCREEN ----------------------------------------
```

Figure A17 ZOC-14 DAS Program: SCAN_ZOC_05

```
660   !
670   CLEAR SCREEN
680   PRINT "Introduction.  Program SCAN_ZOC_05:"
690   PRINT
700   PRINT "  - Scans 1-3 Zoc-14 Modules simultaneously (32 pressure sensing ports each)."
710   PRINT "  - Uses Zero Operate Calibrate (ZOC) principal:"
720   PRINT "       - Collects raw pressure data (Zero Operate)"
730   PRINT "       - Collects calibration data (Calibrate)"
740   PRINT "       - Reduces and stores data on selected hard or floppy drive."
750   PRINT "  - CALSYS2000 Calibration Module used for the reference pressure standard."
760   PRINT "  - Raw pressure data reduced using calibration data from CALSYS2000"
770   PRINT "    and Zocs in the calibration mode."
780   PRINT
790   PRINT "Input variables: Hard and Floppy drive for data storage"
800   PRINT "                 Sample frequency per port (1-50,000 Hz)"
810   PRINT "                 Samples per Port (1-1023)"
820   PRINT "                 Number of Zocs and their capacity"
830   PRINT
840   ! Note: HFS Files limited to 14 characters, LIF Files limited to 10 char.
850   !       Output files have a length of 10 characters to support LIF format.
860   !       Hard drive :,700 is HFS format, :,700,0,1 is LIF format.
870   !       Floppy drive :,700,1 is LIF format.
880   PRINT "Output files:    Raw data =>     ZW(Zoc#)(Date YMMDD)(Run#)"
890   PRINT "                 Calibration =>  ZC(Zoc#)(Date YMMDD)(Run#)"
900   PRINT "                 Reduced data => ZR(Zoc#)(Date YMMDD)(Run#)"
910   DISP "Select F2 key for Key Menu, F3 for system inputs, or F6 for data reduction."
920 Hold: !
930   GOTO Hold
940  !
950 Key_menu: !----- KEY MENU ---------------------------------------------------
960  !
970   CLEAR SCREEN
980   PRINT "ZOC-14 Operating Menu."
990   PRINT
1000  PRINT "Function                                    Function Key"
1010  PRINT
1020  PRINT "  Introduction                                 F1"
1030  PRINT "  Operating Menu                               F2"
1040  PRINT "  System Set-up                                F3"
1050  PRINT "  Data Collection Preparation                  F4"
1060  PRINT "  Data Collection                              F5"
1070  PRINT "  Data Reduction                               F6"
1080  PRINT "  List Files (Copy files to Floppy)            F7"
1090  PRINT
1100  PRINT "  Exit                                         F8"
1110 !
1120  GOTO Hold
1130 !
1140 Input:!----- INPUT VARIABLES ----------------------------------------------
1150  MAT Zoc_cal1= (0)
1160  MAT Zoc_cal2= (0)
1170  MAT Zoc_cal3= (0)
1180  MAT File$= ("-")
1190  Date$=FNDate$(TIMEDATE)
1200  !
1210  CLEAR SCREEN
1220  PRINT "System Set-up."
1230  PRINT
1240  INPUT "Select Hard drive for storing data (0=:,700 1=:,700,0,1)",Drv
1250  IF Drv=0 THEN
1260    Data_drive$=":,700,0,0"
1270  ELSE
1280    Data_drive$=":,700,0,1"
1290  END IF
1300  INPUT "Enter data sampling rate (1-50kHz):",Hz
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
1310  PRINT "Data acquisition rate:";TAB(50);Hz;" Hz"
1320  INPUT "Number of samples per port (1-1023): ",Sample_number
1330  PRINT "Number of samples per port:";TAB(50);Sample_number
1340  INPUT "Number of Zoc's connected to Multi-programer",Zoc_number
1350  PRINT "Number of Zocs to be scanned:";TAB(50);Zoc_number
1360  Cal_mod_id(0)=Zoc_number
1370  FOR Zoc_case=1 TO Zoc_number
1380    SELECT Zoc_case
1390    CASE 1
1400      Run=1
1410      CALL File(1)
1420      INPUT "Enter Calibration Module number set for Zoc #1 (Enter 1 or 2):",Cal_mod_id(1)
1430    CASE 2
1440      Run=1
1450      CALL File(2)
1460      INPUT "Enter Calibration Module number set for Zoc #2 (Enter 1 or 2):",Cal_mod_id(2)
1470    CASE 3
1480      Run=1
1490      CALL File(3)
1500      INPUT "Enter Calibration Module number set for Zoc #3 (Enter 1 or 2):",Cal_mod_id(3)
1510    END SELECT
1520  NEXT Zoc_case
1530  !
1540  Period=1/Hz
1550  Pulse=Period/2                    !Pulse length of HP69736A trigger signal
1560  !
1570  PRINT "Total raw data acquisition time:";TAB(50);Period*Sample_number*31;" sec."
1580  PRINT "Total calibration data acquisition time:";TAB(50);Period*5*31+(7*Pause_for);" sec."
1590  PRINT
1600  PRINT "Data storage disc =>";Data_drive$
1610  PRINT "Data will be stored in the following files beginning with Run #";Run
1620  PRINT
1630  FOR I=1 TO Zoc_number
1640    J=(I-1)*3
1650    PRINT "Raw data file:           ";File$(Run,J+1)
1660    PRINT "Calibration data file:   ";File$(Run,J+2)
1670    PRINT "Reduced data file:       ";File$(Run,J+3)
1680    PRINT
1690  NEXT I
1700  !
1710  DISP "Select F4 key to begin data aquisition"
1720  GOTO Hold
1730  !
1740 Data_prep: !----- PREPARE FOR DATA COLLECTION ----------------------------
1750  CLEAR SCREEN
1760  PRINT "Data Collection Preparation."
1770  PRINT
1780  IF Run=0 THEN
1790    PRINT "Program not initialized for data collection."
1800    DISP "Select F3 to initialize Set-up"
1810    GOTO Hold
1820  END IF
1830  PRINT "Check list:"
1840  PRINT "  - HiScan CALSYS2000 on-line"
1850  PRINT "  - CALMOD supply line valve is OPEN (on back of CALSYS2000)"
1860  PRINT "  - CALSYS2000 (Nitrogen) pressure source at 90 psi"
1870  !
1880  !
1890  CONTROL 9,5;3                     ! Set DTR & RTS to Active for CALSYS2000
1900  OUTPUT 9;VAL$(1);"IC";CHR$(13);END! Initialize Calibrator module #1
1910  OUTPUT 9;VAL$(2);"IC";CHR$(13);END! Initialize Calibrator module #2
1920  WAIT Pause_for                     ! Allow CALSYS2000 to set Zocs
1930  !
1940  DISP "Select F5 to start data aquisition"
1950  GOTO Hold
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
1960 !
1970 Collect_data:!------ COLLECT DATA ----------------------------------------
1980  IF Run=0 THEN
1990    PRINT "Program not initialized for data collection."
2000    DISP "Select F3 to initialize Set-up"
2010    GOTO Hold
2020  END IF
2030  CLEAR SCREEN
2040  PRINT
2050  PRINT "Collecting raw pressure data."
2060  Count=Sample_number*32            ! Set Count as function of sample number
2070                                    ! and number of port readings (32) on
2080                                    ! Zoc for raw data collection.
2090  CALL Scan_zocs(Count,Pulse)      ! Collect raw data into Memory System
2100  PRINT
2110  PRINT "Raw data collection complete."
2120  BEEP
2130 !
2140 Raw_data_xfer:!------ TRANSFER RAW DATA FM MEMORY SYSTEM TO HARD DISC -----
2150  PRINT
2160 !
2170  FOR Zoc_case=1 TO Zoc_number      ! Collect raw data, reduce data and
2180    SELECT Zoc_case                 ! and store reduce data on hard drive
2190    CASE 1
2200      CALL Raw_dat(Buffer1,1)
2210    CASE 2
2220      IF Run>1 THEN
2230        Run=Run-1
2240      END IF
2250      CALL Raw_dat(Buffer2,2)
2260    CASE 3
2270      IF Run>1 THEN
2280        Run=Run-1
2290      END IF
2300      CALL Raw_dat(Buffer3,3)
2310    END SELECT
2320  NEXT Zoc_case
2330 !
2340 Initial_cal:!-------- CALIBRATION SET-UP ---------------------------------
2350 ! Calibration data array for each Zoc: Zoc_cal_(33,10)
2360 ! Format:
2370 !   For ports i=1 to 33
2380 !      Row 0, column 0: Period
2390 !      Row 0, column 1: Sample number
2400 !      Row 0, column 2: Zoc #
2410 !      Row 0, column 3: Calibrator module ID (1=50 psi 2=15 psi)
2420 !      Row 0: __ __ __ __ NH NM NL ZO PL PM PH  (pressure Hg.)
2430 !      Row i: A0 A1 A2 A3 NH NM NL ZO PL PM PH (LS coef,press volts)
2440 !   LS coef are Least Squares curve fit coef for third order polynomial.
2450 !
2460  PRINT
2470  PRINT "Collecting calibration data."
2480  REAL Cal1(1120),Cal2(1120),Cal3(1120)! Calibration data array
2490  Count=32*5                        ! Set count to collect calibration data
2500  !
2510  MAT Zoc_cal1= (0)
2520  MAT Zoc_cal2= (0)
2530  MAT Zoc_cal3= (0)
2540  Zoc_cal1(0,0)=Period
2550  Zoc_cal1(0,1)=Sample_number
2560  Zoc_cal1(0,2)=1
2570  Zoc_cal1(0,3)=Cal_mod_id(1)
2580  Zoc_cal2(0,0)=Period
2590  Zoc_cal2(0,1)=Sample_number
2600  Zoc_cal2(0,2)=2
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
2610  Zoc_cal2(0,3)=Cal_mod_id(2)
2620  Zoc_cal3(0,0)=Period
2630  Zoc_cal3(0,1)=Sample_number
2640  Zoc_cal3(0,2)=3
2650  Zoc_cal3(0,3)=Cal_mod_id(3)
2660  !
2670 Collect_cal_dat:!---- COLLECT RAW CALIBRATION DATA ------------------------
2680 !
2690 ! Collect raw calibration data for each CALSYS2000 setting
2700  FOR Index=1 TO 7
2710    CALL Cal2000(Command_mode$(Index),Index)
2720    CALL Scan_zocs(Count,Pulse)
2730    FOR Zoc_case=1 TO Zoc_number
2740      SELECT Zoc_case
2750      CASE 1
2760        Input_rblock(Buffer1,Cal1(*),160,(Index-1)*160+1)
2770      CASE 2
2780        Input_rblock(Buffer2,Cal2(*),160,(Index-1)*160+1)
2790      CASE 3
2800        Input_rblock(Buffer3,Cal3(*),160,(Index-1)*160+1)
2810      END SELECT
2820    NEXT Zoc_case
2830  NEXT Index
2840  !
2850  ! Store collected calibration data
2860  FOR Zoc_case=1 TO Zoc_number
2870    SELECT Zoc_case
2880    CASE 1
2890      CALL Cal_dat(Cal1(*),Zoc_cal1(*))
2900    CASE 2
2910      CALL Cal_dat(Cal2(*),Zoc_cal2(*))
2920    CASE 3
2930      CALL Cal_dat(Cal3(*),Zoc_cal3(*))
2940    END SELECT
2950  NEXT Zoc_case
2960  !
2970  PRINT
2980  PRINT "Calibration data collection complete."
2990  BEEP
3000  WAIT .25
3010  BEEP
3020  OUTPUT 9;VAL$(1);"IC";CHR$(13);END! Initialize Calibrator module #1
3030  OUTPUT 9;VAL$(2);"IC";CHR$(13);END! Initialize Calibrator module #2
3040  PRINT
3050  PRINT "*** Secure Calibrator pressure valve to conserve Nitrogen ***"
3060  PRINT
3070  PRINT "CALSYS2000 Calibration modes and pressures (in Hg):"
3080 Fmt1:IMAGE /,5X,K,10X,K,10X,K,10X,K
3090  PRINT USING Fmt1;"Mode","Zoc #1","Zoc #2","Zoc #3"
3100 Fmt2:IMAGE 6X,K,10X,3D.4D,8X,3D.4D,8X,3D.4D
3110  FOR I=4 TO 10
3120    PRINT USING Fmt2;Command_mode$(I-3),Zoc_cal1(0,I),Zoc_cal2(0,I),Zoc_cal3(0,I)
3130  NEXT I
3140  DISP "Select F4 for another data run, or F6 to reduce data"
3150  GOTO Hold
3160  !
3170 Reduce_data:!----- REDUCE DATA AND STORE ON HARD DRIVE --------------------
3180 ! Routine loads raw and calibration data from storage drive, reduces the
3190 ! data, and stores the data to the storage drive.
3200 !
3210  CLEAR SCREEN
3220  PRINT "Calibration and Raw data reduction and storage."
3230  PRINT
3240  IF Run=0 THEN
3250    INPUT "Enter the date of data for for reduction (YMMDD):",Date$
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
3260    INPUT "Number of Zoc's connected to Multi-programer",Zoc_number
3270    INPUT "Select data storage drive (0=:,700 1=:,700,0,1 2=:,700,1)",Drv_case
3280    SELECT Drv_case
3290    CASE 0
3300      Data_drive$=":,700,0,0"
3310    CASE 1
3320      Data_drive$=":,700,0,1"
3330    CASE 2
3340      Data_drive$=":,700,1"
3350    END SELECT
3360  END IF
3370  !
3380  MAT File$= ("-")
3390  FOR Zoc_case=1 TO Zoc_number   !Assign files from storage to File$(*)
3400    SELECT Zoc_case
3410    CASE 1
3420      CALL File_scan(1)
3430    CASE 2
3440      CALL File_scan(2)
3450    CASE 3
3460      CALL File_scan(3)
3470    END SELECT
3480  NEXT Zoc_case
3490  !
3500  PRINT "Current files on storage disc ";Data_drive$;" for date ";Date$
3510  PRINT
3520  FOR Rn=1 TO Run
3530    FOR Zn=1 TO Zoc_number
3540      FOR I=1 TO 3
3550        PRINT USING "3X,K,#";File$(Rn,(Zn-1)*3+I)
3560      NEXT I
3570      PRINT USING "+,L"
3580    NEXT Zn
3590  NEXT Rn
3600  PRINT
3610  !
3620  FOR Run_red=1 TO Run              ! Reduce data routine.
3630    FOR Zoc_case=1 TO Zoc_number
3640      SELECT Zoc_case
3650      CASE 1
3660        CALL Raw_red_dat(1,Run_red)
3670      CASE 2
3680        CALL Raw_red_dat(2,Run_red)
3690      CASE 3
3700        CALL Raw_red_dat(3,Run_red)
3710      END SELECT
3720    NEXT Zoc_case
3730  NEXT Run_red
3740  Run=0
3750  Data_reduced=1
3760  BEEP
3770  DISP "Select F3 reinitialize set-up for data collection, or F8 to Exit"
3780  GOTO Hold
3790  !
3800 View_files: !----- VIEW FILES ON STORAGE DRIVE ----------------------------
3810 ! Routine loads files from storage drive and displays file names.
3820 !
3830  CLEAR SCREEN
3840  PRINT "List Raw, Calibration and Reduced data files."
3850  PRINT
3860  IF Data_reduced=1 THEN Print_files
3870  IF Run=0 THEN
3880    INPUT "Enter the date of data for for reduction (YMMDD):",Date$
3890    INPUT "Number of Zoc's connected to Multi-programer",Zoc_number
3900    INPUT "Select data storage drive (0=:,700 1=:,700,0,1 2=:,700,1)",Drv_case
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
3910    SELECT Drv_case
3920    CASE 0
3930      Data_drive$=":,700,0,0"
3940    CASE 1
3950      Data_drive$=":,700,0,1"
3960    CASE 2
3970      Data_drive$=":,700,1"
3980    END SELECT
3990  END IF
4000 Print_files: !
4010  PRINT "Data storage drive name =>  ";Data_drive$
4020 !
4030  MAT File$= ("-")
4040  FOR Zoc_case=1 TO Zoc_number   !Assign files from storage to File$(*)
4050    SELECT Zoc_case
4060    CASE 1
4070      CALL File_scan(1)
4080    CASE 2
4090      CALL File_scan(2)
4100    CASE 3
4110      CALL File_scan(3)
4120    END SELECT
4130  NEXT Zoc_case
4140 !
4150  PRINT
4160  PRINT "Current files on storage disc for date ";Date$
4170  PRINT
4180  FOR Rn=1 TO Run                      !Print the files listing on the
4190    FOR Zn=1 TO Zoc_number             !designated storage drive.
4200      FOR I=1 TO 3
4210        PRINT USING "3X,K,#";File$(Rn,(Zn-1)*3+I)
4220      NEXT I
4230      PRINT USING "/"
4240    NEXT Zn
4250  NEXT Rn
4260  !
4270  IF Drv_case<2 THEN
4280    INPUT "Do you want to copy files from the Hard drive to Floppy? (0=No 1=Yes)",Copy_h_to_f
4290    IF Copy_h_to_f=0 THEN End_view
4300    ON ERROR GOSUB View_error
4310    PRINT
4320    PRINT "WARNING: Any duplicate existing files on the Floppy will be copied over!"
4330    PRINT                              !Copy the files from the designated
4340    FOR Rn=1 TO Run                    !hard drive to the floppy drive.
4350      FOR Zn=1 TO Zoc_number
4360        FOR I=1 TO 3
4370          Fi$=File$(Rn,(Zn-1)*3+I)
4380          COPY Fi$&Data_drive$ TO Fi$&":,700,1"
4390          IF Fi$<>"-" THEN
4400            PRINT "File ";Fi$;" copied to Floppy"
4410          END IF
4420        NEXT I
4430      NEXT Zn
4440    NEXT Rn
4450    PRINT
4460    PRINT "Files have been copied from ";Data_drive$;" to Floppy :,700,1"
4470  END IF
4480  GOTO End_view
4490 View_error:   !
4500  SELECT ERRN
4510  CASE 56                              !File does not exist, then continue.
4520    CLEAR ERROR
4530    ERROR RETURN                       !Return to line following COPY
4540  CASE 54                              !Duplicate file exist on the floppy,
4550    PURGE Fi$&":,700,1"                !then purge the dup file, retrun to
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
4560    CLEAR ERROR
4570    RETURN                                  !the line COPY and copy the file.
4580   CASE ELSE
4590    DISP ERRM$
4600    PAUSE
4610   END SELECT
4620   !
4630   !
4640 End_view: !
4650   Run=0
4660   DISP "Select F2 to return to menu, or F8 to Exit"
4670   GOTO Hold
4680   !
4690 Finish:!
4700   LOAD "ZOC_MENU",10
4710 !
4720   END
4730 End:!=============================================================
4740 ! Function to return todays date for input into file names
4750   DEF FNDate$(Seconds)
4760    Julian=Seconds DIV 86400-1721119
4770    Year=(4*Julian-1) DIV 146097
4780    Julian=(4*Julian-1) MOD 146097
4790    Day=Julian DIV 4
4800    Julian=(4*Day+3) DIV 1461
4810    Day=(4*Day+3) MOD 1461
4820    Day=(Day+4) DIV 4
4830    Month=(5*Day-3) DIV 153   ! Month
4840    Day=(5*Day-3) MOD 153
4850    Day=(Day+5) DIV 5         ! Day
4860    Year=100*Year+Julian
4870    IF Month<10 THEN
4880     Month=Month+3
4890    ELSE
4900     Month=Month+3
4910     Year=Year+1
4920    END IF
4930    Year$=VAL$(Year)
4940    IF Month<10 THEN
4950     Month$="0"&VAL$(Month)
4960    ELSE
4970     Month$=VAL$(Month)
4980    END IF
4990    IF Day<10 THEN
5000     Day$="0"&VAL$(Day)
5010    ELSE
5020     Day$=VAL$(Day)
5030    END IF
5040    D$=Year$[4]&Month$&Day$
5050    RETURN D$
5060   FNEND
5070   !-----------------------------------------------------------------------
5080   ! Subroutine to build file names as required by Run number for a specified
5090   ! Zoc, and assign existing files to the File$ matrix.
5100   SUB File(Zn)
5110    COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),Date$,Run
5120    COM /Files/ File$(*),Data_drive$
5130    DIM Data_disc1$[23],Data_disc2$[23],Data_disc3$[23]
5140    ON ERROR GOTO Error
5150    J=(Zn-1)*3
5160 Assign_file:   !
5170    File1=0
5180    Data_file1$="ZW"&VAL$(Zn)&Date$&VAL$(Run)
5190    Data_disc1$=Data_file1$&Data_drive$
5200    ASSIGN @Check_path1 TO Data_disc1$   !Check for existance of ZW_
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
5210    File$(Run,J+1)=Data_file1$           !Assign ZW_ to matrix.
5220    File1=1                              !Flag to ID file exists.
5230    !
5240    File2=0
5250    Data_file2$="ZC"&VAL$(Zn)&Date$&VAL$(Run)
5260    Data_disc2$=Data_file2$&Data_drive$
5270    ASSIGN @Check_path2 TO Data_disc2$   !Check for existance of ZC_.
5280    File$(Run,J+2)=Data_file2$           !Assign ZC_ to matrix.
5290    File2=1                              !Flag to ID file exists.
5300    !
5310    Data_file3$="ZR"&VAL$(Zn)&Date$&VAL$(Run)
5320    Data_disc3$=Data_file3$&Data_drive$
5330    ASSIGN @Check_path3 TO Data_disc3$   !Check for existance of ZR_.
5340    File$(Run,J+3)=Data_file3$           !Assign ZR_ to matrix.
5350    !
5360    Run=Run+1                            !If ZW_ exist, reassign Run #
5370    ASSIGN @Check_path1 TO *
5380    ASSIGN @Check_path2 TO *
5390    ASSIGN @Check_path3 TO *
5400    GOTO Assign_file                     !Check storage disc again.
5410 Error:  ! Subroutine if ERROR=56, files donot exist for Run and Zoc
5420    IF ERRN<>56 THEN
5430      PRINT ERRM$
5440      PAUSE
5450    END IF
5460    IF File1=0 THEN Fin                  !File ZW_ doesnot exist, exit
5470    IF File1=1 THEN                      !File ZW_ exists
5480      IF File2=0 THEN                    !File ZC_ doesnot exists, therefore
5490        ASSIGN @Check_path1 TO *
5500        PURGE Data_disc1$                !delete ZW_.
5510      ELSE
5520        Run=Run+1                        !File ZW_ & ZC_ exist, step Run
5530      END IF                             !and continue.
5540    END IF
5550    ASSIGN @Check_path1 TO *
5560    ASSIGN @Check_path2 TO *
5570    ASSIGN @Check_path3 TO *
5580    GOTO Assign_file
5590 Fin:   !
5600    ASSIGN @Check_path1 TO *
5610    ASSIGN @Check_path2 TO *
5620    ASSIGN @Check_path3 TO *
5630    Data_file2$="ZC"&VAL$(Zn)&Date$&VAL$(Run)
5640    Data_file3$="ZR"&VAL$(Zn)&Date$&VAL$(Run)
5650    File$(Run,J+1)=Data_file1$           !Create ZW_ to matrix.
5660    File$(Run,J+2)=Data_file2$           !Create ZC_ to matrix.
5670    File$(Run,J+3)=Data_file3$           !Assign ZR_ to matrix.
5680  SUBEND
5690  !-------------------------------------------------------------------------
5700  ! Subroutine to operate the HP6944A Multi-programmer for scanning Zocs.
5710  SUB Scan_zocs(Count,Pulse)
5720    COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Buffer3,Adc3,Timer
5730    Wait_time=Count*2*Pulse+10.0     ! Set Timer wait time to +10 secs.
5740    Init(Timer)                      ! Initialize Timer system
5750    Set_timeout(Timer,Wait_time)     ! Set Pause_for period of xx secs.
5760    Set_count(Timer,Count)           ! Set Count number into Timer
5770    Set_period(Timer,Pulse)          ! Set Timer pulse length in secs.
5780    Init(Buffer1)                    ! Initialize Buffer for data storage
5790    Init(Buffer2)
5800    Init(Buffer3)
5810    Start(Timer)                     ! Start data sample collection
5820    Wait_for(Timer)                  ! Data samples stored in Memory System
5830  SUBEND
5840 !-------------------------------------------------------------------------
5850 ! Subroutine to collect raw pressure data from Memory System and store
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
5860 ! onto the hard drive for future data reduction.
5870  SUB Raw_dat(Buff,Zn)
5880    COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),DateS,Run
5890    COM /Files/ File$(*),Data_drive$ ! Data file listing for 99 runs.
5900    ON ERROR GOTO Error
5910    INTEGER Raw_data(32672) BUFFER  ! Integer raw data buffer for 32*1021
5920                                    ! data samples.  Integer format for
5930                                    ! mininum transfer time to storage.
5940    DIM Data_disc$[23]
5950    Sn=Sample_number
5960 Assign_file:   !
5970    Data_file$=File$(Run,(Zn-1)*3+1) ! Raw data file
5980    Data_disc$=Data_file$&Data_drive$
5990    CREATE BDAT Data_disc$,32*Sn+1,2   ! Create BDAT file of 2 byte records.
6000    ASSIGN @Data_path TO Data_disc$ ! Assign path to hard drive
6010    ASSIGN @Buffer_path TO BUFFER Raw_data(*);FORMAT OFF
6020    Input_iblock(Buff,Raw_data(*),Sn*32,1)     ! Load data samples
6030    CONTROL @Buffer_path,4;32*2*Sn+2       ! Close buffer when full
6040    TRANSFER @Buffer_path TO @Data_path    ! Transfer data Data_disc
6050    ASSIGN @Buffer_path TO *
6060    ASSIGN @Data_path TO *
6070    PRINT "Raw pressure data: Run#";Run;", Zoc#";Zn;", storage drive file ";Data_file$&Data_drive$
6080    GOTO Fin
6090 Error: !
6100    IF ERRN<>54 THEN
6110      PRINT ERRM$
6120      PAUSE
6130    END IF
6140    IF ERRN=54 THEN                     ! Run step routine when compiling
6150      Run=Run+1                         ! multiple data runs without data
6160      CALL File(Zn)                     ! reduction.
6170    END IF
6180    GOTO Assign_file
6190 Fin:     !
6200  SUBEND
6210 !-------------------------------------------------------------------------
6220 ! Subroutine controls calibration mode and reads pressure from Pressure
6230 ! Standard into Zoc_cal(*) array.
6240  SUB Cal2000(Command$,I)
6250    COM /Zoc_dat/ REAL Zoc_cal1(*) BUFFER,Zoc_cal2(*) BUFFER,Zoc_cal3(*) BUFFER
6260    COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),DateS,Run
6270    DIM Pressure$[5]                        ! Required to read data stream
6280    OUTPUT 9;VAL$(1);Command$;CHR$(13);END  ! Sets calibrator #1 mode
6290    OUTPUT 9;VAL$(2);Command$;CHR$(13);END  ! Sets calibrator #2 mode
6300    WAIT Pause_for                          ! Allow CALSYS2000 to stabilize
6310    FOR K=1 TO Cal_mod_id(0)                ! Read CALSYS2000 cal press
6320      SELECT K
6330      CASE 1
6340        OUTPUT 9;VAL$(Cal_mod_id(1));"RP";CHR$(13);END
6350        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal1(0,I+3),Pressure$
6360      CASE 2
6370        OUTPUT 9;VAL$(Cal_mod_id(2));"RP";CHR$(13);END
6380        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal2(0,I+3),Pressure$
6390      CASE 3
6400        OUTPUT 9;VAL$(Cal_mod_id(3));"RP";CHR$(13);END
6410        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal3(0,I+3),Pressure$
6420      END SELECT
6430    NEXT K
6440    IF I<=3 THEN                        ! Account for positive pressures used
6450      Zoc_cal1(0,I+3)=-Zoc_cal1(0,I+3) ! by CALSYS2000 in the NH,NM, & NL mode.
6460      Zoc_cal2(0,I+3)=-Zoc_cal2(0,I+3)
6470      Zoc_cal3(0,I+3)=-Zoc_cal3(0,I+3)
6480    END IF
6490  SUBEND
6500 !-------------------------------------------------------------------------
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
6510 ! Subroutine stores calibration data collected from Memory System and
6520 ! CALSYS2000 calibration pressure data onto the hard drive.
6530 ! Zoc_cal_ is then stored onto the hard drive.
6540  SUB Cal_dat(REAL Cal(*),Zoc_cal(*) BUFFER)
6550    COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),Date$,Run
6560    COM /Files/ File$(*),Data_drive$ ! Data file listing for 99 runs.
6570 !
6580 ! Converting Cal(*) to Zoc_cal(*)
6590    FOR J=4 TO 10                    ! Cal runs: NH,NM,NL,ZO,PL,PM,PH
6600      FOR I=1 TO 32                  ! Zoc ports per calibration run
6610        FOR K=0 TO 4                 ! Number of samples per run
6620          Zoc_cal(I,J)=Zoc_cal(I,J)+Cal(I+K*32+(J-4)*160)
6630        NEXT K
6640        Zoc_cal(I,J)=Zoc_cal(I,J)/5 ! Average of 5 samples per port I
6650      NEXT I
6660    NEXT J
6670 !
6680 ! Transfer calibration data to hard drive.
6690    ON ERROR GOSUB Purge_file
6700    DIM Data_disc$[23]               ! Define string for data file name
6710    Zn=Zoc_cal(0,2)                  ! Define Zoc number
6720    Data_file$=File$(Run,(Zn-1)*3+2) ! Calibration data file
6730    Data_disc$=Data_file$&Data_drive$
6740    CREATE BDAT Data_disc$,33,8*11   ! Create BDAT file of 11*8 byte
6750    ASSIGN @Data_path TO Data_disc$  ! Assign path to hard drive
6760    ASSIGN @Buffer_path TO BUFFER Zoc_cal(*);FORMAT OFF
6770    CONTROL @Buffer_path,4;8*11*33   !Set data file length
6780    TRANSFER @Buffer_path TO @Data_path!Store cal data on hard drive
6790    ASSIGN @Buffer_path TO *         ! Close path
6800    ASSIGN @Data_path TO *           ! Close path
6810    PRINT "Calibration data: Run#";Run;", Zoc#";Zn;", storage drive file ";Data_disc$
6820    GOTO Fin
6830 Purge_file:   !
6840    IF ERRN=54 THEN
6850      PRINT "Error occured in SUB Cal_dat.  Error:";ERRN
6860      PAUSE
6870    END IF
6880    RETURN
6890 Fin:              !
6900  SUBEND
6910  !-------------------------------------------------------------------------
6920  !   broutine loads raw and calibration data from the storage drive,
6930  ! reduces the data, and stores the data onto the storage drive.
6940  ! Calibration data is reduced using the Least Squares Curve fit to obtain
6950  ! coefficients for a third-order polynomial.  The raw pressure data is
6960  ! reduced using these coefficients.
6970  ! Buffer arrays are replaced with standard arrays for data manipulation.
6980  ! Utilization of Buffers and the TRANSFER routine results in lost of the
6990  ! first several data bytes when data is transferred from floppy media to
7000  ! the buffer.  Utilization of OUTPUT, ENTER, and arrays results in no
7010  ! data lost with floppy media.  Hard disc media works well with either
7020  ! data manipulation technique using buffers or standard arrays.
7030  SUB Raw_red_dat(Zn,Rn)
7040    COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Buffer3,Adc3,Timer
7050    COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),Date$[6],Run
7060    COM /Files/ File$(*),Data_drive$   !Data file listing for 99 runs.
7070    Data_file1$=File$(Rn,(Zn-1)*3+2)   ! Calibration data file
7080    Data_file2$=File$(Rn,(Zn-1)*3+1)   ! Raw data file name
7090    Data_file3$=File$(Rn,(Zn-1)*3+3)   ! Reduced data file name
7100    !
7110    IF Data_file3$<>"-" THEN           ! Contin  if Reduce data file
7120      GOTO Fin                         ! doesnot exist.
7130    END IF
7140    !
7150    IF Data_file1$="-" THEN
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
7160      PRINT "Calibration file doesnot exist for Run#";Rn;", Zoc#";Zn
7170      GOTO Fin
7180    END IF
7190    !
7200    ON ERROR GOSUB Error
7210    DIM Data_disc1$[23]
7220    DIM Data_disc2$[23]
7230    DIM Data_disc3$[23]
7240    Data_disc1$=Data_file1$&Data_drive$
7250    REAL Zoc_cal(32,10)               !Array to handle calibration data
7260    !
7270 Data_reduction:   !
7280    PRINT "Data reduction: Run#";Rn;", Zoc#";Zn
7290    !
7300    ASSIGN @Data_path1 TO Data_disc1$;FORMAT OFF
7310    ENTER @Data_path1;Zoc_cal(*)      !Load raw calibration data into array
7320    ASSIGN @Data_path1 TO *
7330    !
7340 ! Calibration data reduction using Least Squares Polynominal fitting.
7350    REAL A(3,3),B(3),C(3),Sum_x(6),A_inv(3,3)! Least Square reduction arrays
7360    FOR K=1 TO 32                 ! Loop for each port
7370 !
7380      MAT C= (0)
7390      MAT Sum_x= (0)
7400 !
7410      FOR J=1 TO 6                ! Routine to reduce individual port cal
7420        FOR I=4 TO 10             ! data into elements to a power x^j
7430          Sum_x(J)=Sum_x(J)+Zoc_cal(K,I)^J
7440        NEXT I
7450      NEXT J
7460 !
7470      FOR I=0 TO 3                ! Derive A array
7480        FOR J=0 TO 3
7490          A(I,J)=Sum_x(I+J)
7500        NEXT J
7510      NEXT I
7520      A(0,0)=7
7530 !
7540      FOR J=0 TO 3                ! Derive C array
7550        FOR I=4 TO 10
7560          C(J)=C(J)+Zoc_cal(K,I)^J*Zoc_cal(0,I)
7570        NEXT I
7580      NEXT J
7590 !
7600      MAT A_inv= INV(A)
7610      MAT B= A_inv*C              ! B array is matrix of Least Square
7620 !                                  coefficients a0,a1,a2,& a3 for polynomial
7630 !                                  equation fitting calibration data for a
7640 !                                  specified port
7650 !
7660 ! Collect Least Square coefficients
7670      Zoc_cal(K,0)=B(0)           !Coefficient a0
7680      Zoc_cal(K,1)=B(1)           !Coefficient a1
7690      Zoc_cal(K,2)=B(2)           !Coefficient a2
7700      Zoc_cal(K,3)=B(3)           !Coefficient a3
7710 !
7720    NEXT K
7730    !
7740    ASSIGN @Data_path1 TO Data_disc1$;FORMAT OFF
7750    OUTPUT @Data_path1;Zoc_cal(*)      !Store reduced calibration data
7760    ASSIGN @Data_path1 TO *
7770    !
7780    PRINT "Calibration data reduced and transferred to ";Data_file1$
7790    !
7800    ! Recover raw data, convert to real, reduce then store in blocks
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
7810    ! of samples (32 ports scanned per block)
7820    Sn=Zoc_cal(0,1)                    !Sample number.
7830    INTEGER Data_integer(1:32)         !Array to handle raw integer data.
7840    REAL Data_real(1:32),Data(32)      !Arrays to handle raw and reduced
7850    Data_disc2$=Data_file2$&Data_drive$ !real data.
7860    Data_file3$="ZR"&VAL$(Zn)&Date$&VAL$(Rn)  !Reduced data file name.
7870    Data_disc3$=Data_file3$&Data_drive$
7880    CREATE BDAT Data_disc3$,Sn,8*33    !BDAT file of 33*8 byte records.
7890    ASSIGN @Data_path2 TO Data_disc2$;FORMAT OFF
7900    ASSIGN @Data_path3 TO Data_disc3$;FORMAT OFF
7910    !
7920    CONTROL @Data_path2,5;2            !Set read pointer to 2nd record
7930    !                                  !in raw interger data file.
7940    FOR Block=1 TO Sn
7950      ENTER @Data_path2;Data_integer(*) !Load raw data into array.
7960      SELECT Zoc_cal(0,2)              !Translating raw interger data into
7970      CASE 1                           !raw real data.
7980        Translate(Adc1,Data_integer(*),Data_real(*),
7990      CASE 2
8000        Translate(Adc2,Data_integer(*),Data_real(*))
8010      CASE 3
8020        Translate(Adc3,Data_integer(*),Data_real(*))
8030      END SELECT
8040    !
8050 ! Routine to reduce raw real data:
8060 !
8070 !       Data = a0 + a1*x + a2*x^2 + a3*x^3
8080 !
8090 ! where a0,a1,a2, & a3 are Least Square coefficients, and x is
8100 ! the individual port raw data value.
8110 !
8120      Data(0)=Block                   ! Store reduce data sample number.
8130      FOR K=1 TO 32
8140        Data(K)=Zoc_cal(K,0)+Zoc_cal(K,1)*Data_real(K)+Zoc_cal(K,2)*Data_real(K)^2+Zoc_cal
8150      NEXT K
8160 !
8170      OUTPUT @Data_path3;Data(*)      !Store block of reduced data into
8180    NEXT Block                        !into the file on the designated drive.
8190    !
8200    ASSIGN @Data_path3 TO *
8210    ASSIGN @Data_path2 TO *
8220    PRINT "Raw data reduced and transferred to ";Data_file3$
8230    PRINT
8240    GOTO Fin
8250 Error:                               !Routine to trap error in program.
8260    PRINT ERRM$
8270    PAUSE
8280    RETURN
8290 Fin:   !
8300  SUBEND
8310 !-------------------------------------------------------------------------
8320  ! Subroutine to load existing files required by Run number for a specified
8330  ! Zoc, and assign existing files to the File$ matrix for Data reduction
8340  ! and List files routines.
8350  SUB File_scan(Zn)
8360  COM /Stats/ REAL Pulse,Sample_number,Pause_for,INTEGER Cal_mod_id(3),Date$,Run
8370    COM /Files/ File$(*),Data_drive$
8380    DIM Data_disc1$[23],Data_disc2$[23],Data_disc3$[23]
8390    Rn=1
8400    Loop=1
8410    File_in_storage=0
8420    ON ERROR GOTO Error
8430    J=(Zn-1)*3
8440    WHILE Loop=1
8450      File1=0
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
8460      Data_file1$="ZW"&VAL$(Zn)&Date$&VAL$(Rn)
8470      Data_disc1$=Data_file1$&Data_drive$
8480      ASSIGN @Check_path1 TO Data_disc1$ !Check for existance of ZW_.
8490      File$(Rn,J+1)=Data_file1$          !Assign ZW_ to matrix.
8500      File1=1
8510    !
8520      Data_file2$="ZC"&VAL$(Zn)&Date$&VAL$(Rn)
8530      Data_disc2$=Data_file2$&Data_drive$
8540      ASSIGN @Check_path2 TO Data_disc2$ !Check for existance of ZC_.
8550      File$(Rn,J+2)=Data_file2$          !Assign ZC_ to matrix.
8560    !
8570      Data_file3$="ZR"&VAL$(Zn)&Date$&VAL$(Rn)
8580      Data_disc3$=Data_file3$&Data_drive$
8590      ASSIGN @Check_path3 TO Data_disc3$ !Check for existance of ZR_.
8600      File$(Rn,J+3)=Data_file3$          !Assign ZR_ to matrix.
8610    !
8620      GOTO Assign_file                   !Check storage disc again.
8630 Error:  ! Subroutine if ERROR=56, files donot exist for Rn and Zoc
8640      IF ERRN<>56 THEN
8650        PRINT ERRM$
8660        PAUSE
8670      END IF
8680 Assign_file:!
8690      IF File1=1 THEN                    !Switch to begin ent ring
8700        File_in_storage=1                !file names in to File$
8710      END IF
8720      IF File1=0 THEN
8730        IF File_in_storage=1 THEN        !Switch t. exit routine
8740          Loop=0                         !once e..es are made
8750        END IF                           !into File$
8760      END IF
8770      ASSIGN @Check_path1 TO
8780      ASSIGN @Check_path2 TO *
8790      ASSIGN @Check_path3 TO *
8800      IF Rn=100 THEN                     !Exit routine after checking
8810        Loop=0                           !up to 100 possible Run #s
8820      END IF
8830      Rn=Rn+1
8840    END WHILE
8850 Fin:   !
8860    Run=Rn-1
8870  SUBEND
8880  !-------------  ------------------------------------------------------------
```

Figure A17 (cont) ZOC-14 DAS Program: SCAN_ZOC_05

```
10   ! Program: READ_ZOC
20   ! Description: Reads specified data compiled from program SCAN_ZOC_04.
30    CLEAR SCREEN
40    PRINTER IS CRT
50    DIM Data_disc1$[23]
60    DIM Data_disc2$[23]
70    INPUT "Enter Zoc # (1,2,3), date (YMMDD), and run #:",Zoc,Date$,Run
80    INPUT "Print results to screen or printer (0=Screen 1=Printer)",View
90    IF View=1 THEN
100     PRINTER IS 711
110   ELSE
120     PRINTER IS CRT
130   END IF
140   Data_file1$="ZC"&VAL$(Zoc)&Date$&VAL$(Run)
150   Data_file2$="ZR"&VAL$(Zoc)&Date$&VAL$(Run)
160   Data_disc1$=Data_file1$&":,700,0,1"
170   Data_disc2$=Data_file2$&":,700,0,1"
180   ASSIGN @Data_path1 TO Data_disc1$
190   ASSIGN @Data_path2 TO Data_disc2$
200   REAL N1,N2
210   STATUS @Data_path1,3;N1              ! Determine number of records
220   STATUS @Data_path2,3;N2              ! Determine number of records
230   ALLOCATE REAL Cal(N1-1,10),Data(1:N2,32)! Define REAL array of records
240   ENTER @Data_path1;Cal(*)
250   ENTER @Data_path2;Data(*)
260   Period=Cal(0,0)
270   Hz=1/Period
280   Sample_number=Cal(0,1)
290   Zoc=Cal(0,2)
300 Print_results: !
310   PRINT "Data Print Out for Zoc #";Zoc;", Run #";Run
320   PRINT
330   PRINT TAB(5);"Period between samples (sec):";Period
340   PRINT TAB(5);"Sample collection rate (Hz): ";Hz
350   PRINT TAB(5);"Number of samples per port:  ";Sample_number
360   PRINT TAB(5);"Length of data run (sec):    ";Period*31*Sample_number
370 Loop:  !
380   PRINT
390   INPUT "Enter port number for data (0=Exit):",Port_number
400   IF Port_number=0 THEN Finish
410   INPUT "Enter First sample# of run to be viewed:",Sample1
420   INPUT "Enter End sample# of run:",Sample2
430   PRINT "Data Tabulation for Port #";Port_number;"from file: ";Data_file2$
440   PRINT
450   PRINT USING "K,6X,K,6X,K";"Sample","Time (sec)","Pressure (Hg.)"
460   FOR Sample=Sample1 TO Sample2
470     PRINT USING "5D,7X,2D.5D,8X,3D.5D";Sample,((Sample-1)*32+(Port_number-1))*Period,Data
480   NEXT Sample
490   GOTO Loop
500 Finish: !
510   ASSIGN @Data_path1 TO *
520   ASSIGN @Data_path2 TO *
530   DEALLOCATE Cal(*)
540   DEALLOCATE Data(*)
550   PRINT
560   PRINT
570   PRINTER IS CRT
580   LOAD "ZOC_MENU",10
590   END
```

Figure A18 ZOC-14 DAS Program: READ_ZOC

```
10    ! Program: PLOT_DATA
20    ! Descript: Plots reduced data compiled by SCAN_ZOC_04.
30    !           Actual data points are ploted as squares.  Square size
40    !           can be adjusted by varying variable Sc in line 110.
50    !
60    COM /Plot_labels/ REAL Xo,Xf,Yo,Yf,Dx,Dy,Title$[60],X_label$[50],Y_label$[50]
70    !
80    CLEAR SCREEN
90    PRINTER IS CRT
100   Sc=.005                                !Scale size of plotting squares
110   PRINT "Program: PLOT_DATA"
120   PRINT
130   PRINT "Program plots the reduced data compiled by the program SCAN_ZOC_05."
140   PRINT "Hard copies of plots can be reproduced to either the HP Think Jet"
150   PRINT "printer or HP Laser Jet printer."
160   PRINT
170   PRINT
180   PRINT "Press F2 to continue after calibration display is plotted"
190   PRINT "Press <Shift> <Dump Graph> for printout of plot on the printer"
200   !
210   ! Load data ------------------------------------------------------------
220   DIM Data_disc1$[23]
230   DIM Data_disc2$[23]
240   INPUT "Enter Zoc # (1,2,3), Date (YMMDD), Run #:",Zoc,Date$,Run
250   File_name1$="ZC"&VAL$(Zoc)&Date$&VAL$(Run)
260   File_name2$="ZR"&VAL$(Zoc)&Date$&VAL$(Run)
270   Data_disc1$=File_name1$&":,700,0,1"
280   Data_disc2$=File_name2$&":,700,0,1"
290   ASSIGN @Data_path1 TO Data_disc1$
300   ASSIGN @Data_path2 TO Data_disc2$
310   REAL N1,N2
320   STATUS @Data_path1,3;N1              ! Determine number of records
330   STATUS @Data_path2,3;N2              ! Determine number of records
340   ALLOCATE REAL Cal(N1-1,10),Data(1:N2,32)! Define REAL array of records
350   ENTER @Data_path1;Cal(*)
360   ENTER @Data_path2;Data(*)
370   Period=Cal(0,0)
380   Hz=1/Period
390   Sample_number=Cal(0,1)
400   Zoc=Cal(0,2)
410 Display_stats: !-------------------------------------------------------------
420   CLEAR SCREEN
430   PRINT "Program plots reduced data from file ";File_name2$
440   PRINT
450   PRINT "Statistics for Zoc #";Zoc
460   PRINT
470   PRINT TAB(5);"Period between samples (msec):";Period*1000
480   PRINT TAB(5);"Sample collection rate (Hz): ";Hz
490   PRINT TAB(5);"Number of samples per port:  ";Sample_number
500   PRINT TAB(5);"Length of data run (msec):    ";((Sample_number-1)*32+31)*Period*1000
510   PRINT
520   PRINT
530   PRINT "Data point can be plotted as a continuous line, or squares."
540   PRINT
550   PRINT "Note: Wait for symbol '*' in lower right corner of CRT to change"
560   PRINT "      to a '-' before pressing <Shift><Dump Graph>"
570   !
580   INPUT "Enter Port #:",P
590   INPUT "Enter min range of pressure reading (in Hg):",Yo
600   INPUT "Enter max range of pressure reading (in Hg):",Yf
610   INPUT "Enter Start time for plot (msec):",Xo
620   INPUT "Enter Stop time for plot (msec)):",Xf
630   INPUT "Enter type of data point plotting (0=Squares 1=Line):",Plot_case
640   INPUT "Plots DUMPED to Think Jet or Laser Jet: (0=TJ 1=LJ)",Dump_device
650   IF Dump_device=1 THEN
```

Figure A19 ZOC-14 DAS Program: PLOT_DATA

```
660     DUMP DEVICE IS 9
670   ELSE
680     DUMP DEVICE IS 711
690   END IF
700   !
710   Title$="Reduced Data Plot of Port# "&VAL$(P)
720   X_label$="Time (msec)"
730   Y_label$="Pressure (In Hg)"
740   Dx=10                              !Tic marks per X-axis on plot
750   Dy=10                              !Tic marks per Y-axis on plot
760   !
770   ! Plot routine ------------------------------------------------------------
780   CALL Plot                            !Rountine to display graph background
790   FOR I=1 TO Sample_number             !Plot Squares around data points
800     Sample_time=((Data(I,0)-1)*32+(P-1))*1000*Period
810     SELECT Plot_case
820     CASE 0
830       PLOT Sample_time,Data(I,P),-2    !Move pen to data point position
840       CALL Square(Xo,Xf,Yo,Yf,Sc)      !Subrountine to plot squares
850     CASE 1
860       PLOT Sample_time,Data(I,P)
870     END SELECT
880   NEXT I
890     !
900   PAUSE
910 Finish: !-----------------------------------------------------------------
920   CLEAR SCREEN
930   INPUT "Graph another calibration plot: ( 0=No 1=Yes )",Again
940   IF Again=1 THEN Display_stats
950   ASSIGN @Data_path1 TO *
960   ASSIGN @Data_path2 TO *
970   DEALLOCATE Cal(*)
980   DEALLOCATE Data(*)
990   PRINTER IS CRT
1000  DUMP DEVICE IS 711
1010  LOAD "ZOC_MENU",10
1020  END
1030  !=================================================================
1040  SUB Plot
1050  !
1060  ! Subroutine to display plot screen, less the plots of any curves
1070  ! for the specified variables in the COM /Plot_labels/ line.
1080    COM /Plot_labels/ Xo,Xf,Yo,Yf,Dx,Dy,Title$,X_label$,Y_label$
1090    CLEAR SCREEN
1100    KEY LABELS OFF
1110    GINIT                            !Initialize graph routine
1120    X_range=Xf-Xo                    !Length of X-axis
1130    Y_range=Yf-Yo                    !Length of Y-axis
1140    LORG 6                           !Character ref pt:top center
1150    MOVE 100*RATIO/2,100             !Move cursor to screen loc for labels
1160    CSIZE 3                          !Sizes labeling
1170    LABEL Title$                     !Plot title
1180    MOVE 100*RATIO/2,0               !Move cursor to bottom center screen
1190    LORG 4                           !Character ref pt:bottom center
1200    LABEL X_label$                   !X-axis label
1210    DEG                              !Desig degrees for LDIR
1220    LDIR 90                          !Sets Y-axis label on end
1230    LORG 6
1240    MOVE 0,50
1250    LABEL Y_label$                   !Y-axis label
1260    LDIR 0                           !Reset label to horizontal orientation.
1270    LORG 2                           !Chr ref pt:left center
1280    VIEWPORT 10,90*RATIO,10,90       !Sets graph screen size
1290    FRAME                            !Box around VIEWPORT
1300    WINDOW Xo,Xf,Yo,Yf               !Set axis lengths in VIEWPORT
```

Figure A19 (cont) ZOC-14 DAS Program: PLOT_DATA

```
1310    AXES X_range/Dx,Y_range/Dy,Xo,Yo      !Axes intersect at lower left
1320    AXES X_range/Dx,Y_range/Dy,Xf,Yf      !Axes intersect at upper right
1330    GRID X_range/Dx,Y_range/Dy,Xo,Yo,Dx,Dy,.001
1340    CLIP OFF                          !So labels can print outside VIEWPORT
1350    CSIZE 3.0,.4                      !Axes label size
1360    LORG 6                            !Number X-axis
1370    FOR I=Xo TO Xf STEP X_range/Dx
1380      MOVE I,Yo-.01*Y_range
1390      LABEL USING "#,K";I
1400    NEXT I
1410    LORG 8                            !Number Y-axis
1420    FOR I=Yo TO Yf STEP Y_range/Dy
1430      MOVE Xo-.01*X_range,I
1440      LABEL USING "#,K";I
1450    NEXT I
1460    CLIP ON
1470  !
1480  SUBEND
1490  SUB Square(Xo,Xf,Yo,Yf,Sc)
1500  !Subroutine to plot squares around the local origin designated
1510  !by the PLOT statement.
1520    Xd=Sc*(Xf-Xo)                     !X displacement for RPLOT
1530    Yd=Sc*(Yf-Yo)*RATIO               !Y displacement for RPLOT
1540    RPLOT -Xd,Yd,-2
1550    RPLOT Xd,Yd,-1
1560    RPLOT Xd,-Yd,-1
1570    RPLOT -Xd,-Yd,-1
1580    RPLOT -Xd,Yd,2
1590  SUBEND
```

Figure A19 (cont) ZOC-14 DAS Program: PLOT_DATA

```
10    ! Program: CAL_READ_PR1
20    ! Description: program to operate CAL2000 calibration modes sequentially
30    !              and read corresponding calibration pressures in Hg as a
40    !              test bed for later insertion into the Zoc scanning programs
50    ! Author: Rick Wendland, Naval Postgraduate School, Monterey CA
60    !         Tele:(408) 646-2165
70    !
80    CLEAR SCREEN
90    PRINTER IS CRT
100   PRINT "Program: CAL_READ_PR1"
110   PRINT
120   PRINT "   This program sequentially sets the CAL2000 calibration"
130   PRINT "   modes and reads the corresponding internal Pressure Standard"
140   PRINT "   for that mode."
150   PRINT
160   PRINT "          PH            Positive high range pressure to CAL(+)"
170   PRINT "          PM            Positive mid range pressure to CAL(+)"
180   PRINT "          PL            Positive low range pressure to CAL(+)"
190   PRINT "          ZO            CAL(+) & REF(-) connected together"
200   PRINT "          NL            Negative low range pressure to REF(-)"
210   PRINT "          NM            Negative mid range pressure to REF(-)"
220   PRINT "          NH            Negative high range pressure to REF(-)"
230   !
240   INPUT "Display results to CRT or PRINTER? (0=CRT 1=PRINTER)",Results
250   IF Results=1 THEN
260     PRINTER IS 711
270   END IF
280   CONTROL 9,5;3           !Set DTR & RTS to active for CAL2000 configuration
290   INTEGER Error,Value
300   REAL Pressure(1:7)
310   DIM Command_mode$(1:7)[2],Pressure$[5]
320   Command_mode$(1)="NH"
330   Command_mode$(2)="NM"
340   Command_mode$(3)="NL"
350   Command_mode$(4)="ZO"
360   Command_mode$(5)="PL"
370   Command_mode$(6)="PM"
380   Command_mode$(7)="PH"
390   ON ERROR GOTO Find_error
400   OUTPUT 9;"IC";CHR$(13);END              !Initialize CAL2000
410   WAIT 1.5
420 Send_command:  !
430   FOR I=1 TO 7
440     OUTPUT 9;Command_mode$(I);CHR$(13);END
450     WAIT 1.5                        !Wait time to allow calibrator stabilization
460     OUTPUT 9;"RP";CHR$(13);END
470     ENTER 9 USING "#,SD.5DESZZ,K";Pressure(I);Pressure$
480     GOTO No_error
490 Find_error:  !
500     STATUS 9,10;Error
510     STATUS 9,6;Value
520     Error_code$=IVAL$(Error,2)
530     PRINT "Register 10:       ";Error_code$[9,16]
540     PRINT "Register  6:       ";Value,CHR$(Value)
550 No_error:!
560    !DISP "F2 TO CONTINUE"       !Steps used to determine wait time between
570    !PAUSE                       !CALMOD steps.  ZOC output connected to the
580   NEXT I                        !O-scope, use stopwatch to measure settling
590 Print_results:!                 !time of dc voltage from ZOC.
600   PRINT
610   PRINT
620   PRINT "CAL2000: Calibration modes and pressures."
630   PRINT
640   PRINT USING "2X,K,5X,K";"Mode","Pressure (in Hg)"
650   PRINT
```

Figure A20 ZOC-14 DAS Program: CAL_READ_PR1

```
660   FOR I=1 TO 7
670     PRINT USING "3X,K,10X,3D.4D";Command_mode$(I),Pressure(I)
680   NEXT I
690 Finish:  !
700   PRINTER IS CRT
710   DISP "Press F2 to continue"
720   PAUSE
730   LOAD "ZOC_MENU",10
740   END
```

Figure A20 (cont) ZOC-14 DAS Program: CAL_READ_PR1

```
10   ! Program: TABULATE_ZOC
20   ! Description: Tabulates data compiled from program SCAN_ZOC_04.
30    DIM Data_disc1$[23]
40    DIM Data_disc2$[23]
50 Input:  !
60    CLEAR SCREEN
70    PRINT "Program tabulates Zoc pressures and calibration data from"
80    PRINT "the SCAN_ZOC_05 program."
90    INPUT "Zoc # (1,2,3), date (YMMDD), and run #  of data to be reviewed:",Zoc,Date$,Run
100   CLEAR SCREEN
110   Data_file1$="ZC"&VAL$(Zoc)&Date$&VAL$(Run)
120   Data_file2$="ZR"&VAL$(Zoc)&Date$&VAL$(Run)
130   Data_disc1$=Data_file1$&":,700,0,1"
140   Data_disc2$=Data_file2$&":,700,0,1"
150   ASSIGN @Data_path1 TO Data_disc1$
160   ASSIGN @Data_path2 TO Data_disc2$
170   REAL N1,N2
180   STATUS @Data_path1,3;N1              ! Determine number of records
190   STATUS @Data_path2,3;N2              ! Determine number of records
200   ALLOCATE REAL C(N1-1,10)            ! Define REAL array of records
210   ALLOCATE REAL D(1:N2,32)            ! Define REAL array of records
220   ENTER @Data_path1;C(*)
230   ENTER @Data_path2;D(*)
240  !
250   INPUT "First port of calibration data to be displayed (0=Exit):",Port_o
260   IF Port_o=0 THEN Finish
270   INPUT "Last port of calibration data to be displayed:",Port_f
280   INPUT "Print results to CRT or Printer (0=CRT 1=Printer)?",Prt
290   IF Prt=0 THEN
300     PRINTER IS CRT
310   ELSE
320     PRINTER IS 711
330   END IF
340 Print_results: !
350   PRINT "Reduced Data Tabulation at a sample rate of";1/C(0,0);" Hz"
360   PRINT
370 Format1:IMAGE K,2X,K,2X,K,2X,K
380   PRINT USING Format1;"Port","Sample 1","Sample 2","Sample 3"
390 Format2:IMAGE 4D,2X,4D.3D,2X,4D.3D,2X,4D.3D
400   I=1                              !ID Data for a given sampling number
410   FOR P=Port_o TO Port_f
420     PRINT USING Format2;P,D(I,P),D(I+1,P),D(I+2,P)
430   NEXT P
440   PRINT
450   PRINT "Calibration Data Tabulation for Zoc#";Zoc
460   PRINT
470   PRINT "Pressure voltage readings:"
480 Format3:IMAGE K,4X,K,6X,K,6X,K,6X,K,6X,K,6X,K,6X,K
490   PRINT USING Format3;"Port","NH","NM","NL","ZO","PL","PM","PH"
500 Format4:IMAGE 3D,X,3D.3D,X,3D.3D,X,3D.3D,X,3D.3D,X,3D.3D,X,3D.3D,X,3D.3D,X,3D.3D
510   PRINT USING Format4;0,C(0,4),C(0,5),C(0,6),C(0,7),C(0,8),C(0,9),C(0,10)
520   FOR P=Port_o TO Port_f
530     PRINT USING Format4;P,C(P,4),C(P,5),C(P,6),C(P,7),C(P,8),C(P,9),C(P,10)
540   NEXT P
550   PRINT
560   PRINT "Calibration polynomial coefficients for Zoc#";Zoc
570 Format5:IMAGE K,8X,K,16X,K,16X,K,16X,K
580   PRINT USING Format5;"Port","A0","A1","A2","A3"
590   FOR P=Port_o TO Port_f
600     PRINT P;TAB(5);C(P,0);TAB(24);C(P,1);TAB(43);C(P,2);TAB(62);C(P,3)
610   NEXT P
620   !
630   ASSIGN @Data_path1 TO *
640   ASSIGN @Data_path2 TO *
650   DEALLOCATE C(*)
```

Figure A21 ZOC-14 DAS Program: TABULATE_ZOC

```
660   DEALLOCATE D(*)
670   !
680   INPUT "Review data or Exit (0=Exit 1=Review):",Action
690   IF Action=0 THEN Finish
700   PRINT
710   GOTO Input
720 Finish: !
730   PRINTER IS CRT
740   LOAD "ZOC_MENU",10
750   END
```

Figure A21 (cont) ZOC-14 DAS Program: TABULATE_ZOC

```
10    ! Program: LS_PLOT
20    ! Descript: Plots calibration results and Least Square curve from
30    !           data provided by  SCAN_ZOC_04
40    !
50    COM /Plot_labels/ REAL Xo,Xf,Yo,Yf,Dx,Dy,Title$[60],X_label$[50],Y_label$[50]
60    ! Input variables ----------------------------------------------------------
70    CLEAR SCREEN
80    PRINT "Program: LS_PLOT"
90    PRINT
100   PRINT "Program plots the calibration curve from Least Square coefficients"
110   PRINT "and the calibration data points contain in the Calibration Data File."
120   PRINT
130   PRINT
140   PRINT "Press F2 to continue after calibration display is plotted"
150   PRINT "Press <Shift> <Dump Graph> for printout of plot on the printer"
160   INPUT "Enter Calibration Data file name:",Data_file$
170 Loop:  !
180   INPUT "Enter Port #:",P
190   INPUT "Enter min range of pressure reading (in Hg):",Xo
200   INPUT "Enter max range of pressure reading (in Hg):",Xf
210   INPUT "Enter min range of Zoc voltage reading (Volts):",Yo
220   INPUT "Enter max range of Zoc voltage reading (Volts):",Yf
230   INPUT "Plots DUMPED to Think Jet or Laser Jet Printer:(0=TJ 1=LJ)",Dump_device
240   IF Dump_device=1 THEN
250     DUMP DEVICE IS 9
260   ELSE
270     DUMP DEVICE IS 711
280   END IF
290   !
300   Title$="Least Square Plot of CAL2000 Calibration (Port# "&VAL$(P)&" )"
310   Y_label$="Volts"
320   X_label$="Pressure (In Hg)"
330   Dx=10                                 !Tic marks per X-axis on plot
340   Dy=12                                 !Tic marks per Y-axis on plot
350   CLEAR SCREEN
360   !
370   ! Load data ----------------------------------------------------------------
380   DIM Data_disc$[23]
390   Data_disc$=Data_file$&":,700,0,1"     !Define data transfer path
400   ASSIGN @Data_path TO Data_disc$
410   REAL N
420   STATUS @Data_path,3;N                 !Determine length of data file
430   ALLOCATE REAL D(N-1,10)               !Allocate an array to accept data
440   ENTER @Data_path;D(*)                 !Load data into memory in array D(*)
450   !
460   ! PLot routine -------------------------------------------------------------
470   CALL Plot                             !Rountine to display graph background
480   FOR X=-3 TO 3 STEP .1                 !Plot calibration curve
490     Fx=(D(P,0)+D(P,1)*X+D(P,2)*X^2+D(P,3)*X^3)
500     PLOT Fx,X
510   NEXT X
520   Sc=.01                                !Scale size of plotting squares
530   FOR I=4 TO 10                         !Plot Squares around data points
540     PLOT D(0,I),D(P,I),-2               !Move pen to data point position
550     CALL Square(Xo,Xf,Yo,Yf,Sc)         !Subrountine to plot squares
560   NEXT I
570   ASSIGN @Data_path TO *
580   !
590   PAUSE
600   CLEAR SCREEN
610   INPUT "Graph another calibration plot: ( 0=No 1=Yes )",Again
620   DEALLOCATE D(*)
630   IF Again=1 THEN Loop
640   LOAD "ZOC_MENU",10
650   END
```

Figure A22 ZOC-14 DAS Program: LS_PLOT

```
!=============================================================
SUB Plot
! Subroutine to display plot screen, less the plots of any curves
! for the specified variables in the COM /Plot_labels/ line.
  COM /Plot_labels/ Xo,Xf,Yo,Yf,Dx,Dy,Title$,X_label$,Y_label$
  CLEAR SCREEN
  KEY LABELS OFF
  GINIT                              !Initialize graph routine
  X_range=Xf-Xo                      !Length of X-axis
  Y_range=Yf-Yo                      !Length of Y-axis
  LORG 6                             !Character ref pt:top center
  MOVE 100*RATIO/2,100               !Move cursor to screen loc for labels
  CSIZE 3                            !Sizes labeling
  LABEL Title$                       !Plot title
  MOVE 100*RATIO/2,0                 !Move cursor to bottom center screen
  LORG 4                             !Character ref pt:bottom center
  LABEL X_label$                     !X-axis label
  DEG                                !Desig degrees for LDIR
  LDIR 90                            !Sets Y-axis label on end
  LORG 6
  MOVE 0,50
  LABEL Y_label$                     !Y-axis label
  LDIR 0                             !Reset label to horizontal orientation.
  LORG 2                             !Chr ref pt:left center
  VIEWPORT 10,90*RATIO,10,90         !Sets graph screen size
  FRAME                              !Box around VIEWPORT
  WINDOW Xo,Xf,Yo,Yf                 !Set axis lengths in VIEWPORT
  AXES X_range/Dx,Y_range/Dy,Xo,Yo       !Axes intersect at lower left
  AXES X_range/Dx,Y_range/Dy,Xf,Yf       !Axes intersect at upper right
  GRID X_range/Dx,Y_range/Dy,Xo,Yo,Dx,Dy,.001
  CLIP OFF                           !So labels can print outside VIEWPORT
  CSIZE 3.0,.4                       !Axes label size
  LORG 6                             !Number X-axis
  FOR I=Xo TO Xf STEP X_range/Dx
    MOVE I,Yo-.01*Y_range
    LABEL USING "#,K";I
  NEXT I
  LORG 8                             !Number Y-axis
  FOR I=Yo TO Yf STEP Y_range/Dy
    MOVE Xo-.01*X_range,I
    LABEL USING "#,K";I
  NEXT I
  CLIP ON
!
SUBEND
SUB Square(Xo,Xf,Yo,Yf,Sc)
!Subroutine to plot squares around the local origin designated
!by the PLOT statement.
  Xd=Sc*(Xf-Xo)                        !X displacement for RPLOT
  Yd=Sc*(Yf-Yo)*RATIO                  !Y displacement for RPLOT
  RPLOT -Xd,Yd,-2
  RPLOT Xd,Yd,-1
  RPLOT Xd,-Yd,-1
  RPLOT -Xd,-Yd,-1
  RPLOT -Xd,Yd,2
SUBEND
```

Figure A22 (cont) ZOC-14 DAS Program: LS_PLOT

```
10    !Program: ZOC_MENU
20    !Description:  Menu for utilization of the Zoc-14 and CALSYS 2000.
30    CLEAR SCREEN
40    CONTROL CRT,5;4
50    KEY LABELS ON
60    ON KEY 1 LABEL "SCAN 1-3ZOCS" GOTO Scan_1
70    ON KEY 2 LABEL "READ ZOCDATA" GOTO Read_z
80    ON KEY 3 LABEL "PLOT ZOCDATA" GOTO Plot
90    ON KEY 4 LABEL "READ    CALSYS20" GOTO Calsys
100   ON KEY 5 LABEL "TABULATECAL DATA " GOTO Tab_data
110   ON KEY 6 LABEL "PLOT CALDATA" GOTO Plot_cal
120   ON KEY 7 LABEL "HP6944A MENU" GOTO Main
130   ON KEY 8 LABEL "EXIT    MENU" GOTO Exit
140   !
150   PRINT "Zoc Electronic Pressure Module Operation Menu"
160   PRINT
170   PRINT "Item:                              Select Function Key"
180   PRINT
190   PRINT "  Scan 1-3 ZOC-14 Modules (32 ports ea)    F1"
200   PRINT "  Read reduced data from ZOC-14 module     F2"
210   PRINT "  Plot reduced data from ZOC-14 module     F3"
220   PRINT "  Read CALSYS 2000 calibration pressures   F4"
230   PRINT "  Read tabulated calibration data          F5"
240   PRINT "  Plot Calibration data                    F6"
250   PRINT
260   PRINT "  HP6944A Main Menu                        F7"
270   PRINT "  Exit Menu                                F8"
280   !
290 Hold:    !
300   GOTO Hold
310 Scan_1:  !
320   CLEAR SCREEN
330   PRINT "Loading SCAN_ZOC_05"
340   LOAD BIN "/WORKSTATIONS/BIN5.1/TRANS"
350   LOAD BIN "/WORKSTATIONS/BIN5.1/SERIAL"
360   LOAD "SCAN_ZOC_05",10
370 Plot:  !
380   LOAD BIN "/WORKSTATIONS/BIN5.1/GRAPHX"
390   LOAD "PLOT_DATA",10
400 Read_z:  !
410   LOAD "READ_ZOC",10
420 Calsys:  !
430   LOAD BIN "/WORKSTATIONS/BIN5.1/SERIAL"
440   LOAD "CAL_READ_PR1",10
450 Tab_data:  !
460   LOAD "TABULATE_ZOC",10
470 Plot_cal:  !
480   LOAD BIN "/WORKSTATIONS/BIN5.1/GRAPHX"
490   LOAD "LS_PLOT",10
500 Main:    !
510   LOAD "HP6944A_MENU",10
520 Exit:CLEAR SCREEN
530   END
```

Figure A23 ZOC-14 DAS Program: ZOC_MENU

APPENDIX B. ZOC-14 PROGRAM DEVELOPMENT CHRONOLOGY

The following prargraphs provide a chronological summary of the SCAN_ZOC program development using the HP14753A CAT Programs. First, the HP6944A manuals required the hardware configuration of the I/O cards be determined. Second, review of the interface requirements to operate the ZOC-14 units identified the binary ZOC address codes to set. The HP69730A Relay Output Card then provided the closure contacts required to set the required address codes[28]. The HP69759A A/D Card provided the signal conversion process to read the ZOC-14. I/O card edge connectors are fabricated from hardware supplied with the I/O cards to provide the interface between the cards and the ZOC-14. The program "SCAN_ZOC_01", Figure B1, provides control and reads data voltage values from the ZOC-14. SCAN_ZOC_01 was the simplest in the series of development programs.

"SCAN_ZOC_02" , Figure B2, integrated new programming features and hardware into a complete data acquisition program for utilizing one ZOC-14. The significant features were,

- segmenting the main program into "blocks" of routines
- utilizing arrays for data processing

---

[28] Refer to Ref. 26 for details on the Relay Output Card.

- hardware integration of the CALSYS2000
- integration of the Buffer and Timer Functions
- data reduction using Least Squares calibration curve fitting
- data storage to the hard drive using BUFFER arrays.

This program did not utilize the Zero Operate and Calibrate method. Calibration data were first collected and stored in a BUFFER array. Raw pressure data were then collected and retained in the HP69791A Memory Card. Calibration data were reduced using the Least Squares routine. The calibration data were then transferred to the hard drive calibration data file. Raw pressure data were extracted from the memory card, reduced, and stored on the hard drive reduced-pressure data file.

BUFFER arrays were used for the high speed data handling capabilities. A data transfer was initiated with a buffer and the computer executed the next line without waiting for the transfer to complete. This process allowed data transfer while the computer was performing the next command, thereby minimizing processing time.

SCAN_ZOC_02 did not control the CALSYS2000. Control was provided by a seperate 386 desk top computer with a modem connected to the CALSYS2000 through the computer's RS-232C port. Interfacing the HP9000 with the CALSYS2000 required resolution of RS-232C data stream handling techniques.

"SCAN_ZOC_03", Figure B3, introduced new features. Over SCAN_ZOC_02, the new features were;

- the use of subroutines and the CASE statements to provide multiple ZOC-14 interfacing into the data acquisition system.
- incorporating full control of the CALSYS2000 using HP9000 commands.

The techniques to control the CALSYS2000 are discussed in section II.B.2.d.

"SCAN_ZOC_04", Figure B4, incorporated the following additional features,

- incorporation of a second ZOC-14 into the process to validate multiple ZOC-14 operation
- command lines to handle a second CALMOD 2000
- incorporation of the Zero Operate and Calibrate principle.

The command line to operate a second CALMOD did not interfere with the current CALSYS2000 operation since this command used an address character "2" and the installed CALMOD only responded to address character "1".

Integration of the "Zero Operate and Calibrate" principle required collecting measured pressure data followed by calibration data. This process posed a memory storage problem for multiple ZOCs. The problem was rectified by transferring the raw data to a hard drive data file for recall off-line. Three advantages were associated with this technique:

1. Programming simplification, by not having to identify raw and calibration data separately on the memory card.
2. Additional data collection capacity was made available by sequential utilization of the memory card.
3. Multiple data and calibration runs could be conducted which were limited only by the hard drive storage capacity.

Data reduction could be accomplished at any later time, providing considerable flexibility. It is to be noted, however, that sufficient space on the hard drive was required to be set aside to store the reduced data. The user must manage hard drive space when acquiring large quantities of data. Transfer of raw and calibration data to the floppy disk is one solution. This would require the user to generate a transfer program similar to the FILE_XFER program, Figure D17, on the /UTILity directory.

The Scan Frequency input (1-100,000 Hz) for the SCAN_ZOC_02, SCAN_ZOC_03, and SCAN_ZOC_04 programs was improperly implemented. The Timer Function required that a specified "Period" be input to define the length of the "step" or logic-high portion of the square wave trigger output[29]. The programs previous to SCAN_ZOC_05 used the reciprocal of the Scan Frequency as the Period to be input. This resulted in square wave periods twice the intended value. SCAN_ZOC_05 correctly adjusts the Scan Frequency input used to define the

---

[29] Ref. 15 defines the input variables for the Timer Function.

Timer Function Period value, changing the input frequency range to "1-50,000 Hz". In SCAN_ZOC_05 the Scan Frequency input value is first inverted, then the resulting period is divided by two to obtain the Timer Period value. In the first three programs this division by two was omitted and thus the programs gave a two-fold decrease in the data sampling rate. This error was discovered during the development of the PLOT_DATA program, Figure A20. The program plots reduced pressure data versus time for a specified ZOC port number. The detected error was detected when the plot showed only half the number of data points for the calculated time span. That SCAN_ZOC_05 uses the correct Timer Period value was verified by the PLOT_DATA program.

```
10   ! Program: SCAN_ZOC_01
20   ! Description: Application program to operate HP6944A collecting voltage
30   !              readings from a ZOC using HP6944A cards.
40   !              Program provides voltage reading from a single ZOC for a
50   !              specified port number 1-32.
60   !              This program was primarily an instruction aid to learn about
70   !              HP6944A operation and ZOC operation.
80   !
90   ! Hardware:  HP6944A Multi-processor
100  !            -(1) Relay Output Card (HP69730A)
110  !            -(1) 500 kHz A/D Card (HP69759A)
120  !            (1) ZOC-14 32 port Electronic Pressure Scanning Module
130  !
140  ! Notes: 1.  ZOC address driven by Relay Output card connecting ZOC's
150  !            A0-A4 leads to the Relay Output card's edge connector:
160  !
170  !                  ZOC         Relay Contact
180  !                  A0               A
190  !                  A1               C
200  !                  A2               E
210  !                  A3               H
220  !                  A4               K
230  !
240  !               Relay Contacts 1,3,5,7, & 9 were connected together and
250  !               tied into ZOC's ground wire to complete the address circuit.
260  !               (Ref ZOC-14 Instruction Manual dwg 8329 sht 3 of 3)
270  !            2. ZOC's Output Plus lead was connected to A/D edge card
280  !               connection W, and ZOC Output Common connected to A/D Y
290  !               (Ref HP69759A Manual pg 2-3, fig. 2-1)
300  !            3. ZOC powered by OA-2 Op Amp Designer internal power supply.
310  !            4. CALSYS2000 sets the different ZOC modes through a pneumatic
320  !               switch matrix.  CALSYS2000 defaults to the Zero Operate mode
330  !               when energized providing CTL2 (Px CTL) control pressure to the
331  !               ZOC, and allow senor pressure readings of any of the 32 ports.
340  !            5. A by-pass valve is installed between CALSYS2000 PSC and
350  !               CALMOD to minimized Nitrogen usage by closing the valve.
360  !               CALMOD is not operative until external ASCII commands are
370  !               provided via RS_232C cable to the CALMOD following CALSYS2000
380  !               energizing.
390  !
400  ! LOADSUB ALL FROM "CONFIGURE"          !These two commands must be initated
410  ! LOADSUB ALL FROM "LIBRARY5.1"         !to append required SUB and CSUB
420  !                                       !routines to the end of this program
430   COM /Issscom/ INTEGER X(1:1106)
440   COM /Isss_heap/ Isss_heap(1000)
450   COM /Names/ Zoc_address,Zoc_pressure  !Identify: Relay Output, A/D
460   Configure("Ask_me","ZOC_CONFIG_01")   !Load Configuration File, and
470  !                                      !request for file review/alteration.
480   System_init                           !Initialize Relay Output and A/D
490  !
500  ! ------ BODY OF APPLICATION PROGRAM ----------------------------------------
510  !
520  REAL Port_reading,Port_req
530  INPUT "Enter Port # (1-32): ",Port_req !Input port #
540  Output(Zoc_address,32-Port_req)        !Select port # by Relay Output
550  Input(Zoc_pressure,Port_reading)       !Read ZOC voltage value from A/D
560  !
570  PRINT "Port #";Port_req;" reading: ";Port_reading
580  !
590   STOP
600  ! --------- END OF MAIN PROGRAM -----------------------------------------
610  !
620   END
```

Figure B1 Development Program: SCAN_ZOC_01

```
10   ! Program: SCAN_ZOC_02
20   ! Description: Application program to operate HP6944A collecting pressure
30   !              readings from one ZOC-14 32 port module using the CAL2000
40   !              to provide calibration data, reduce raw pressure data and
50   !              store data to the hard drive.
60   ! Hardware:  HP6944A Multi-processor
70   !            -500 kHz A/D Card (HP69759A)
80   !            -High Speed Memory Card (HP69791A)
90   !            -Timer/Pacer Card (HP69736A)
100  !            -Counter Card (HP69775A)
110  !            CAL2000 Calibration Module
120  !            ZOC-14 32 port Electronic Pressure Scanning Module
130  ! Note: This program utilizes only one (1) Zoc Module.
140  !
150  ! Buffer Memory: 65536 16-bit data words in HP69791A
160  ! Timer: Maximum 32678 counts for one HP69775A
170  !
180   COM /Isssom/ INTEGER X(1:1106)
190   COM /Isss_heap/ Isss_heap(1000)
200   COM /Names/ Relay,Buffer,Adc1,Timer
210   Configure("Menu_off","ZOC_CONFIG_02")
220  !Configure("Ask_me","ZOC_CONFIG_02")
230  !
240 Body: ! ------ BODY OF PROGRAM ----------------------------------------
250  !
260 Input_variables: !----- INITIAL PARAMETERS --------------------------------
270   PRINT "Program:"
280   PRINT "  - Scans a single Zoc-14 module with 32 pressure sensing ports."
290   PRINT "  - Stores reduced data on the hard drive (:,700,0,1)."
300   PRINT "  - CAL2000 Calibration Module used for the reference pressure standard."
310   PRINT "  - Raw pressure data reduced using calibration data from CAL2000"
320   PRINT "    and data from Zocs in the calibration mode."
330   PRINT
340   PRINT "Input variables: Scan frequency (1-100,000 Hz)"
350   PRINT "                 Samples per Port (1-1021)"
360  !
370  ! Max speed of HP system is Period=0.000002 sec. or 500 kHz.
380  !
390   PRINT
400   INPUT "Enter data rate (1-100kHz):",Hz
410   INPUT "Number of samples per port (1-1021): ",Sample_number
420   Period=1/Hz
430   PRINT
440   PRINT "Data acquisition rate:  ";1/Period;"  Hz"
450   PRINT "Number of samples per port: ",Sample_number
460   PRINT "Total raw data acquisition time: ",Period*Sample_number*32;" sec."
470  !
480 Cal_initial: !----- CALIBRATION SET-UP ------------------------------------
490  !
500   REAL Cal(1:1120)                  ! Calibration collection array for
510                                     ! 5 samples per port, 7 calibration
520                                     ! runs using CAL2000.
530   REAL Zoc_cal(32,10)               ! Calibration data array.  Format:
540   Zoc_cal(0,0)=Period
550   Zoc_cal(0,1)=Sample_number
560  !
570  !   For ports i=1 to 32
580  !      Row 0: 0 0 0 0 NH NM NL ZO PL PM PH  (pressure Hg.)
590  !      Row i: A0 A1 A2 A3 NH NM NL ZO PL PM PH (LS coef,press volts)
600  !   LS coef are Least Squares curve fit coef for third order polynomial.
610  !
620   Calibrate=1                       ! Set calibration run
630   Count=160                         ! Set count to collect calibration data
640   Index=1                           ! Set index for cal data collection
650   PRINT
```

Figure B2 Development Program: SCAN_ZOC_02

```
660   PRINT "Ensure CAL2000 is on-line and initialized"
670   PRINT
680   PRINT "Set CAL2000 calibrate mode (sequence: NH,NM,NL,ZO,PL,PM,PH)"
690   PRINT
700 Cal_routine: !-------------------------------------------------------------
710   INPUT "Read RP value, and enter:",Zoc_cal(0,Index+3)
720   PRINT "Secondary Pressure Standard: ";Zoc_cal(0,Index+3)
730  !
740 Scan_zocs: !---------------------------------------------------------------
750  !Note: Zoc address driven by Counter Card binary output at edge connection.
760  ! Calibration routine collects 5 samples per Zoc port.
770  ! Raw data collection follows calibration collection routine.
780  !
790   Wait_time=INT(Count*Period)+10     ! Set Timer wait time to +10 secs.
800   Init(Timer)                        ! Initialize Timer system
810   Set_timeout(Timer,Wait_time)       ! Set Wait_for period of xx secs.
820   Set_count(Timer,Count)             ! Set Count number into Timer
830   Set_period(Timer,Period)           ! Set Timer pulse length in secs.
840   Init(Buffer)                       ! Initialize Buffer for data storage
850   Start(Timer)                       ! Start data sample collection
860   Wait_for(Timer)                    ! Data samples stored in Memory System
870  !
880  ! Collect calibration data from Memory System
890   IF Calibrate=0 THEN Reduce_cal_dat
900   Input_rblock(Buffer,Cal(*),160,(Index-1)*160)! Collect cal run data
910   Index=Index+1
920   IF Index<8 THEN Cal_routine        ! Loop cal data collection runs
930  !
940  ! Collect raw data initialization
950   CLEAR SCREEN
960   PRINT "Reset Zoc mode to read pressure data"
970   DISP "Press F2 to continue"
980   PAUSE
990   CLEAR SCREEN
1000  PRINT "Collecting raw pressure data." ! Set parameters to collect raw data
1010  Calibrate=0
1020  Count=Sample_number*32             ! Set Count as function of sample number
1030                                     ! and number of port readings (32) on
1040                                     ! Zoc for raw data collection.
1050  GOTO Scan_zocs                     ! Raw data run.
1060 !
1070 ! End of Zoc scan routine
1080 Reduce_cal_dat:!----- REDUCE CALIBRATION DATA ----------------------------
1090 ! Routine to reduce Cal(*) into Zoc_cal(I,J)
1100 !
1110  REAL A(3,3),B(3),C(3),Sum_x(6),A_inv(3,3) ! Least Square reduction arrays
1120  PRINT
1130  PRINT "Reducing calibration data."
1140  FOR J=4 TO 10                      ! Cal runs: NH,NM,NL,ZO,PL,PM,PH
1150    FOR I=1 TO 32                    ! Zoc ports per calibration run
1160      FOR K=0 TO 4                   ! Number of samples per run
1170        Zoc_cal(I,J)=Zoc_cal(I,J)+Cal(I+K*32+(J-4)*160)
1180      NEXT K
1190      Zoc_cal(I,J)=Zoc_cal(I,J)/5    ! Average of 5 samples per port I
1200    NEXT I
1210  NEXT J
1220 !
1230 Least_squares:!
1240 ! Calibration data reduction using Least Squares Polynominal fitting.
1250 !
1260  FOR K=1 TO 32                ! Loop for each port
1270 !
1280    MAT C= (0)
1290    MAT Sum_x= (0)
1300 !
```

Figure B2 (cont) Development Program: SCAN_ZOC_02

```
1310    FOR J=1 TO 6                ! Routine to reduce individual port cal
1320      FOR I=4 TO 10             ! data into elements to a power x^j
1330        Sum_x(J)=Sum_x(J)+Zoc_cal(K,I)^J
1340      NEXT I
1350    NEXT J
1360 !
1370    FOR I=0 TO 3                ! Derive A array
1380      FOR J=0 TO 3
1390        A(I,J)=Sum_x(I+J)
1400      NEXT J
1410    NEXT I
1420    A(0,0)=7
1430 !
1440    FOR J=0 TO 3                ! Derive C array
1450      FOR I=4 TO 10
1460        C(J)=C(J)+Zoc_cal(K,I)^J*Zoc_cal(0,I)
1470      NEXT I
1480    NEXT J
1490 !
1500    MAT A_inv= INV(A)
1510    MAT B= A_inv*C              ! B array is matrix of Least Square
1520 !                                coefficients a0,a1,a2,& a3 for polynomial
1530 !                                equation fitting calibration data for a
1540 !                                specified port
1550 !
1560    Zoc_cal(K,0)=B(0)          ! Collect Least Square coefficients
1570    Zoc_cal(K,1)=B(1)
1580    Zoc_cal(K,2)=B(2)
1590    Zoc_cal(K,3)=B(3)
1600 !
1610  NEXT K
1620 !
1630 Data_transfer:!----- TRANSFER DATA FM MEMORY SYSTEM TO HARD DISC ----------
1640 ! Routine transfers data from Memory System to hard drive via buffer blocks
1650 !
1660 ! Transfer calibration data and coefficients to hard drive
1670  DIM Data_disc1$[23]               ! Define string for data file name
1680  Data_file1$="ZOC_CAL"
1690  Data_disc1$=Data_file1$&":,700,0,1"
1700  PURGE Data_disc1$
1710  CREATE BDAT Data_disc1$,33,8*11   ! Create BDAT file of 11*8 byte
1720  ASSIGN @Data_path1 TO Data_disc1$ ! Assign path to hard drive
1730  OUTPUT @Data_path1;Zoc_cal(*)     ! x33 records and store on hard drive
1740  ASSIGN @Data_path2 TO *            ! Close buffer path
1750 !
1760 ! Reducing raw data and transferring data to hard drive
1770  PRINT
1780  PRINT "Reducing raw pressure data and transferring data to hard drive."
1790  INTEGER Data_int(0:32)            ! Extracted INTEGER raw data
1800  REAL Data_raw(0:32)               ! Translated REAL raw data
1810  REAL Data_red(0:32)               ! REAL reduced data
1820  REAL Data_buffer2(0:32) BUFFER    ! Raw data transfer buffer
1830  REAL Data_buffer3(0:32) BUFFER    ! Reduced data transfer buffer
1840  DIM Data_disc2$[23]               ! Define string for data file name
1850  DIM Data_disc3$[23]
1860  REAL Nd
1870  Data_file2$="ZOC_RAW"
1880  Data_file3$="ZOC_REDUCE"
1890  Data_disc2$=Data_file2$&":,700,0,1"
1900  Data_disc3$=Data_file3$&":,700,0,1"
1910  PURGE Data_disc2$
1920  PURGE Data_disc3$
1930  CREATE BDAT Data_disc2$,10,8*33   ! Create BDAT file of 33*8 byte records
1940                                    ! and initial 10 records
1950                                    ! Each record contains one scan of the
```

Figure B2 (cont) Development Program: SCAN_ZOC_02

```
1960                                    ! 32 port Zoc in REAL pressure values
1970                                    ! corrected by CAL2000 values
1980  CREATE BDAT Data_disc3$,10,8*33
1990  ASSIGN @Data_path2 TO Data_disc2$  ! Assign path to hard drive
2000  ASSIGN @Data_path3 TO Data_disc3$
2010  Count_inblock=32                  ! Number of samples per block transfered
2020  Block_number=Count/Count_inblock  ! Number of transfer blocks
2030 !
2040  FOR Block=1 TO Sample_number      ! Loop routine to transfer data
2050    ASSIGN @Buffer_path2 TO BUFFER Data_buffer2(*);FORMAT OFF
2060    ASSIGN @Buffer_path3 TO BUFFER Data_buffer3(*);FORMAT OFF
2070    Input_iblock(Buffer,Data_int(*),Count_inblock,1)! Load data samples to
2080                                                    ! buffer in blocks
2090    Translate(Adc1,Data_int(*),Data_raw(*))! Load INTEGER data into
2100                                           ! REAL data buffer array
2110 !
2120 ! Routine to reduce raw data using polynomial:
2130 !
2140 !   Data_reduce = a0 + a1*x + a2*x^2 + a3*x^3
2150 !
2160 ! where a0,a1,a2, & a3 are Least Square coefficients, and x is
2170 ! the individual port raw data value.
2180 !
2190    FOR K=1 TO 32
2200      Data_red(K)=Zoc_cal(K,0)+Zoc_cal(K,1)*Data_raw(K)+Zoc_cal(K,2)*Data_raw(K)^2+Zoc_cal(K,3)*Data_raw(K)^3
2210    NEXT K
2220 !
2230    Data_raw(0)=Period*(Block-1)        ! Store raw data sample time
2240    Data_red(0)=Period*(Block-1)        ! Store reduce data sample time
2250 !
2260    MAT Data_buffer2= Data_raw          ! Transfer raw & red data to buffer
2270    MAT Data_buffer3= Data_red
2280 !
2290    CONTROL @Buffer_path2,4;(Count_inblock+1)*8! Close REAL buffer when full
2300    CONTROL @Buffer_path3,4;(Count_inblock+1)*8
2310    TRANSFER @Buffer_path2 TO @Data_path2! Transfer data to hard drive
2320    TRANSFER @Buffer_path3 TO @Data_path3
2330    ASSIGN @Buffer_path2 TO *
2340    ASSIGN @Buffer_path3 TO *
2350  NEXT Block
2360 !
2370  STATUS @Data_path3,3;Nd            ! Determine data file length
2380  ASSIGN @Data_path2 TO *            ! Close buffer path
2390  ASSIGN @Data_path3 TO *
2400 !
2410  CLEAR SCREEN
2420  PRINT "Calibration data file: ";Data_file1$
2430  PRINT "Raw data file: ";Data_file2$
2440  PRINT "Reduced data:  ";Data_file3$
2450  PRINT "Number of records (scans) in ";Data_file3$;" is: ";Nd
2460  PRINT
2470  PRINT "Use READ_ZOC to read data from ";Data_file2$;" and ";Data_file3$
2480  PRINT "Use TABULATE_ZOC to read calibration data from ";Data_file1$
2490  !
2500 Finish:!
2510  DISP "Press F2 to continue."
2520  PAUSE
2530  LOAD "HP6944A_MENU",10
2540 End_body:! --------- END OF MAIN PROGRAM ---------------------------------
2550 !
2560  END
```

Figure B2 (cont) Development Program: SCAN_ZOC_02

```
10   ! Program: SCAN_ZOC_03
20   ! Description: Application program to operate HP6944A collecting pressure
30   !              readings from 1-3 ZOC-14 32 port modules using the CAL2000
40   !              to provide calibration data, reduce raw pressure data and
50   !              store data to the hard drive.
60   ! Hardware:  HP6944A Multi-processors
70   !            -500 kHz A/D Cards (HP69759A)
80   !            -High Speed Memory Cards (HP69791A)
90   !            -Timer/Pacer Card (HP69736A)
100  !            -Counter Card (HP69775A)
110  !            CAL2000 Calibration Module
120  !            ZOC-14 32 port Electronic Pressure Scanning Modules
130  ! Notes: 1. This program utilizes up to three (3) Zoc Modules storing data
140  !        of each Zoc into a seperate buffer Memory System (HP69791A).
150  !        2. COM /Names line and BDAT file ZOC_CONFIG_03 must match for this
160  !        program to operate.
170  !
180  ! Buffer Memory: 65536 16-bit data words in HP69791A per system
190  ! Timer: Maximum 32676 counts for one HP69775A
200  ! Max speed of HP system is Period=0.000002 sec. or 500 kHz.
210  !
220   COM /Isssbcom/ INTEGER X(1:1106)
230   COM /Isss_heap/ Isss_heap(1000)
240   COM /Names/ Buffer,Adc1,Timer
250   Configure("Menu_off","ZOC_CONFIG_03")
260  !Configure("Ask_me","ZOC_CONFIG_03")
270  !
280 Input: !----- INPUT VARIABLES ---------------------------------------------
290   PRINT "Program:  SCAN_ZOC_03"
300   PRINT "  - Scans 1-3 Zoc-14 Modules (32 pressure sensing ports each)."
310   PRINT "    Selected number of Zocs determines the size of stored data files."
320   PRINT "  - CAL2000 Calibration Module used for the reference pressure standard."
330   PRINT "  - Raw pressure data reduced using calibration data from CAL2000"
340   PRINT "    and data from Zocs in the calibration mode."
350   PRINT "  - Stores reduced data on the hard drive (:,700,0,1)."
360   PRINT "  - Program designed to operate three (3) Memory-A/D Cards simultaneously."
370   PRINT
380   PRINT "Input variables: Scan frequency (1-100,000 Hz)"
390   PRINT "                 Samples per Port (1-1021)"
400   PRINT "                 Number of Zocs to be used"
410   PRINT "                 Calibration/Reduced data file name ID"
420  !
430  ! COM assigns calibration data array for 3x32 Zoc ports (96 total)
440   COM /Zoc_dat/ REAL Zoc_cal(96,10) BUFFER
450   COM /Stats/ REAL Period,Sample_number
460   PRINT
470   INPUT "Enter data rate (1-100kHz):",Hz
480   INPUT "Number of samples per port (1-1021): ",Sample_number
490   INPUT "Number of Zoc's connected to Multi-programer",Zoc_number
500   Period=1/Hz
510   Wait_for=1.5                        ! Wait time for CAL2000 stabilization
520   PRINT
530   PRINT "Data acquisition rate:";TAB(50);Hz;" Hz"
540   PRINT "Number of samples per port:";TAB(50);Sample_number
550   PRINT "Number of Zocs to be scanned:";TAB(50);Zoc_number
560   PRINT "Total number of ports to be scanned:";TAB(50);Zoc_number*32
570   PRINT "Total calibration data acquisition time:";TAB(50);Period*5*32+(9*Wait_for);" sec."
580   PRINT "Total raw data acquisition time:";TAB(50);Period*Sample_number*32;" sec."
590   !
600  ! Calibration data array: Zoc_cal(96,10)
610  ! Format:
620  !   For ports i=1 to 96
630  !      Row 0, column 0: Period
640  !      Row 0, column 1: Sample number
650  !      Row 0, column 2: Number of Zocs being used
```

Figure B3 Development Program: SCAN_ZOC_03

```
660  !       Row 0:  __ __ __ __ NH NM NL ZO PL PM PH  (pressure Hg.)
670  !       Row i: A0 A1 A2 A3 NH NM NL ZO PL PM PH (LS coef,press volts)
680  !    LS coef are Least Squares curve fit coef for third order polynomial.
690  !
700   Zoc_cal(0,0)=Period
710   Zoc_cal(0,1)=Sample_number
720   Zoc_cal(0,2)=Zoc_number
730   !
740   PRINT
750   PRINT "Ensure CAL2000 is on-line, calibration pressure source at 90 psi,"
760   PRINT "and calibrator pressure cut-off valve is open (on back of CAL2000)"
770   DISP "Press F2 to start data aquisition"
780   PAUSE
790  !
800 Initial_cal: !-------- CALIBRATION SET-UP ---------------------------------
810  !
820   CLEAR SCREEN
830   PRINT "Collecting calibration data."
840   REAL Cal1(1120),Cal2(1120),Cal3(1120)! Calibration data array
850   REAL Cal(1120)                       ! SUB prog call data array
860   Count=32*5                           ! Set count to collect calibration data
870   CONTROL 9,5;3                        ! Set DTR & RTS to active for CAL2000
880   DIM Command_mode$(1:7)[2]
890   Command_mode$(1)="NH"
900   Command_mode$(2)="NM"
910   Command_mode$(3)="NL"
920   Command_mode$(4)="ZO"
930   Command_mode$(5)="PL"
940   Command_mode$(6)="PM"
950   Command_mode$(7)="PH"
960   OUTPUT 9;"IC";CHR$(13);END           ! Initializes CAL2000
970   WAIT Wait_for                        ! Allows CAL2000 pressure to stabilize
980   !
990 Collect_cal_dat: !---- COLLECT RAW CALIBRATION DATA ------------------------
1000 !
1010 ! Collect raw calibration data for each CAL2000 setting
1020  FOR Index=1 TO 7
1030    CALL Cal2000(Command_mode$(Index),Index,Wait_for)
1040    CALL Scan_zocs(Count,Period)
1050    FOR Zoc_case=1 TO Zoc_number
1060      SELECT Zoc_case
1070      CASE 1
1080        CALL Zoc_dat_cal(Buffer,Cal1(*),Index)
1090      CASE 2
1100        CALL Zoc_dat_cal(Buffer2,Cal2(*),Index)
1110      CASE 3
1120        CALL Zoc_dat_cal(Buffer3,Cal3(*),Index)
1130      END SELECT
1140    NEXT Zoc_case
1150  NEXT Index
1160  !
1170  ! Adjust Calibration pressure sign to account for CAL2000/Zoc method
1180  ! of setting negative calibration pressures.
1190  Zoc_cal(0,4)=-Zoc_cal(0,4)           ! NH adjustment
1200  Zoc_cal(0,5)=-Zoc_cal(0,5)           ! NM adjustment
1210  Zoc_cal(0,6)=-Zoc_cal(0,6)           ! NL adjustment
1220  !
1230  CLEAR SCREEN
1240  PRINT "Calibration data collection complete."
1250  PRINT
1260  PRINT "*** Secure Calibrator pressure valve to conserve Nitrogen ***"
1270  PRINT
1280  PRINT "CAL2000 Calibration modes and pressures (in Hg):"
1290  PRINT TAB(5);"NH";TAB(15);Zoc_cal(0,4)
1300  PRINT TAB(5);"NM";TAB(15);Zoc_cal(0,5)
```

Figure B3 (cont) Development Program: SCAN_ZOC_03

```
1310  PRINT TAB(5);"NL";TAB(15);Zoc_cal(0,6)
1320  PRINT TAB(5);"ZO";TAB(15);Zoc_cal(0,7)
1330  PRINT TAB(5);"PL";TAB(15);Zoc_cal(0,8)
1340  PRINT TAB(5);"PM";TAB(15);Zoc_cal(0,9)
1350  PRINT TAB(5);"PH";TAB(15);Zoc_cal(0,10)
1360  !
1370 Collect_raw_dat: !--- COLLECT RAW PRESSURE DATA --------------------------
1380  PRINT
1390  PRINT "Collecting raw pressure data." ! Set parameters to collect raw data
1400  OUTPUT 9;"IC";CHR$(13);END          ! Reinitialize CAL2000
1410  WAIT Wait_for
1420  Count=Sample_number*32              ! Set Count as function of sample number
1430                                      ! and number of port readings (32) on
1440                                      ! Zoc for raw data collection.
1450  CALL Scan_zocs(Count,Period)        ! Collect raw data into Memory System
1460  PRINT
1470  PRINT "Raw data collection complete."
1480 !
1490 Reduce_cal_dat:!----- REDUCE CALIBRATION DATA ----------------------------
1500 ! Routine to reduce Cal_(*) into Zoc_cal(I,J)
1510 !
1520  PRINT
1530  PRINT "Reducing calibration data."
1540  FOR Zoc_case=1 TO Zoc_number
1550    SELECT Zoc_case
1560    CASE 1
1570      CALL Zoc_dat_red(Cal1(*),1)
1580    CASE 2
1590      CALL Zoc_dat_red(Cal2(*),2)
1600    CASE 3
1610      CALL Zoc_dat_red(Cal3(*),3)
1620    END SELECT
1630  NEXT Zoc_case
1640 !
1650 Data_transfer:!----- TRANSFER DATA FM MEMORY SYSTEM TO HARD DISC ----------
1660 ! Routine transfers data from Memory System to hard drive via buffer blocks
1670 !
1680  CLEAR SCREEN
1690 ! Transfer calibration data and coefficients to hard drive
1700  ON ERROR GOSUB Purge_file
1710  INPUT "Enter Calibration and Reduced Data file name ID (11 char max):",File_id$
1720  Hard_drive$=":,700,0,1"
1730  REAL Nd,Nr
1740 !
1750 !---Transfer calibration data to hard drive
1760  PRINT "Transferring calibration data to the hard drive."
1770  DIM Data_disc1$[23]                 ! Define string for data file name
1780  Data_file1$="ZC_"&File_id$          ! Zoc calibration file name
1790  File_name$=Data_file1$              ! Used for file purging if required
1800  Data_disc1$=Data_file1$&Hard_drive$
1810  CREATE BDAT Data_disc1$,33,8*11     ! Create BDAT file of 11*8 byte
1820  ASSIGN @Data_path1 TO Data_disc1$ ! Assign path to hard drive
1830  ASSIGN @Buffer_path1 TO BUFFER Zoc_cal(*) !Assign buffer path
1840  CONTROL @Buffer_path1,4;8*11*(Zoc_number*32+1) !Set data file length
1850  TRANSFER @Buffer_path1 TO @Data_path1  !Store cal data on hard drive
1860  STATUS @Data_path1,3;Nr             ! Number of records in cal file
1870  ASSIGN @Buffer_path1 TO *           ! Close path
1880  ASSIGN @Data_path1 TO *             ! Close path
1890 !
1900 !---Reducing raw data and transferring data to hard drive
1910  PRINT
1920  PRINT "Reducing raw pressure data and transferring data to the hard drive."
1930  DIM Data_disc2$[23]                 ! Define string for data file name
1940  Data_file2$="ZR_"&File_id$          ! Reduced data file name
1950  File_name$=Data_file2$              ! Used for file purging if required
```

Figure B3 (cont) Development Program: SCAN_ZOC_03

```
1960  Data_disc2$=Data_file2$&Hard_drive$
1970  CREATE BDAT Data_disc2$,10,8*33   ! Create BDAT file of 33*8 byte records
1980                                    ! and initial 10 records
1990                                    ! Each record contains one scan of the
2000                                    ! 32 port Zoc in REAL pressure values
2010                                    ! corrected by CAL2000 values
2020  ASSIGN @Data_path2 TO Data_disc2$  ! Assign path to hard drive
2030 !
2040  FOR Zoc_case=1 TO Zoc_number      ! Collect raw data, reduce data and
2050    SELECT Zoc_case                 ! and store reduce data on hard drive
2060    CASE 1
2070      CALL Raw_red_dat(Buffer,1,@Data_path2)
2080    CASE 2
2090      CALL Raw_red_dat(Buffer2,2,@Data_path2)
2100    CASE 3
2110      CALL Raw_red_dat(Buffer3,3,@Data_path2)
2120    END SELECT
2130  NEXT Zoc_case
2140 !
2150  STATUS @Data_path2,3;Nd            ! Reduced data file length
2160  ASSIGN @Data_path2 TO *            ! Close buffer path
2170 !
2180  CLEAR SCREEN
2190  PRINT "Calibration data file: ";Data_file1$;" containing";Nr;" records."
2200  PRINT "Reduced data file:     ";Data_file2$;" containing";Nd;" records."
2210  PRINT
2220  PRINT "Files ";Data_file1$;" and ";Data_file2$;" are located on hard drive ";Hard_drive$
2230  PRINT "Data reading programs available from HP6944A menu."
2240  !
2250 Finish:!
2260  DISP "Press F2 to continue."
2270  PAUSE
2280  LOAD "HP6944A_MENU",10
2290 Purge_file: !
2300  IF ERRN=54 THEN
2310    PURGE File_name$&Hard_drive$
2320  END IF
2330  RETURN
2340 !
2350  END
2360 End:!=============================================================
2370 ! Routine to operate CAL2000
2380  SUB Cal2000(Command$,I,Wait_period)
2390    COM /Zoc_dat/ REAL Zoc_cal(96,10) BUFFER
2400    DIM Pressure$[5]                ! Required to read data stream
2410    OUTPUT 9;Command$;CHR$(13);END  ! Sets calibration mode
2420    WAIT Wait_period                ! Allow CAL2000 to stabilize
2430    OUTPUT 9;"RP";CHR$(13);END      ! Reads CAL2000 calibration pressure
2440    ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal(0,I+3),Pressure$
2450  SUBEND
2460 !-------------------------------------------------------------------------
2470 ! Routine to operate HP6944A to collect pressure data and store in Memory
2480 !Note: -Zoc address driven by Counter Card binary output at edge connection
2490 !       through a 7404 open collector DTL/TTL located in an auxiliary box.
2500 !
2510  SUB Scan_zocs(Count,Period)
2520    COM /Names/ Buffer,Adc1,Timer
2530    Wait_time=INT(Count*Period)+10  ! Set Timer wait time to +10 secs.
2540    Init(Timer)                     ! Initialize Timer system
2550    Set_timeout(Timer,Wait_time)    ! Set Wait_for period of xx secs.
2560    Set_count(Timer,Count)          ! Set Count number into Timer
2570    Set_period(Timer,Period)        ! Set Timer pulse length in secs.
2580    Init(Buffer)                    ! Initialize Buffer for data storage
2590    !INIT(BUFFER2)
2600    !INIT(BUFFER3)
```

Figure B3 (cont) Development Program: SCAN_ZOC_03

```
2610    Start(Timer)                    ! Start data sample collection
2620    Wait_for(Timer)                 ! Data samples stored in Memory System
2630  SUBEND
2640 !-------------------------------------------------------------------------
2650 ! Routine to read raw calibration data from Memory System
2660  SUB Zoc_dat_cal(Buff,Cal(*),I)
2670    Input_rblock(Buff,Cal(*),160,(I-1)*160+1)
2680  SUBEND
2690 !-------------------------------------------------------------------------
2700 ! Routine to reduce Cal_(*) into Zoc_cal(I,J)
2710  SUB Zoc_dat_red(Cal(*),Zoc)
2720    COM /Zoc_dat/ REAL Zoc_cal(96,10) BUFFER
2730    REAL A(3,3),B(3),C(3),Sum_x(6),A_inv(3,3)! Least Square reduction arrays
2740    REAL Calx(256,5)
2750    FOR J=5 TO 10                   ! Cal runs: NH,NM,NL,ZO,PL,PM,PH
2760      FOR I=1 TO 32                 ! Zoc ports per calibration run
2770        I1=(Zoc-1)*32+I
2780        FOR K=0 TO 4                ! Number of samples per run
2790          Zoc_cal(I1,J)=Zoc_cal(I1,J)+Cal(I+K*32+(J-4)*160)
2800          Calx(I+(J-4)*32,K+1)=Cal(I+K*32+(J-4)*160)
2810        NEXT K
2820        Zoc_cal(I1,J)=Zoc_cal(I1,J)/5 ! Average of 5 samples per port I
2830      NEXT I
2840    NEXT J
2850 !
2860    GOTO Least_squares
2870 ! Print raw calibration data
2880    PRINTER IS 711
2890    PRINT "Raw calibration data for Zoc$";Zoc
2900    PRINT
2910 Format:IMAGE 3D,2X,3D.3D,2X,3D.3D,2X,3D.3D,2X,3D.3D,2X,3D.3D
2920    FOR K=1 TO 7
2930      PRINT "RP=";K
2940      FOR J=1 TO 32
2950        I=J+((K-1)*32)
2960        PRINT USING Format;J,Calx(I,1),Calx(I,2),Calx(I,3),Calx(I,4),Calx(I,5)
2970      NEXT J
2980    NEXT K
2990    PRINTER IS CRT
3000 !
3010 Least_squares:!
3020 ! Calibration data reduction using Least Squares Polynominal fitting.
3030 !
3040    FOR K=1 TO 32                   ! Loop for each port
3050      K1=(Zoc-1)*32+K
3060 !
3070      MAT C= (0)
3080      MAT Sum_x= (0)
3090 !
3100      FOR J=1 TO 6                  ! Routine to reduce individual port cal
3110        FOR I=4 TO 10               ! data into elements to a power x^j
3120          Sum_x(J)=Sum_x(J)+Zoc_cal(K1,I)^J
3130        NEXT I
3140      NEXT J
3150 !
3160      FOR I=0 TO 3                  ! Derive A array
3170        FOR J=0 TO 3
3180          A(I,J)=Sum_x(I+J)
3190        NEXT J
3200      NEXT I
3210      A(0,0)=7
3220 !
3230      FOR J=0 TO 3                  ! Derive C array
3240        FOR I=4 TO 10
3250          C(J)=C(J)+Zoc_cal(K1,I)^J*Zoc_cal(0,I)
```

Figure B3 (cont) Development Program: SCAN_ZOC_03

```
3260        NEXT I
3270      NEXT J
3280 !
3290      MAT A_inv= INV(A)
3300      MAT B= A_inv*C           ! B array is matrix of Least Square
3310 !                               coefficients a0,a1,a2,& a3 for polynomial
3320 !                               equation fitting calibration data for a
3330 !                               specified port
3340 !
3350      Zoc_cal(K1,0)=B(0)       ! Collect Least Square coefficients
3360      Zoc_cal(K1,1)=B(1)
3370      Zoc_cal(K1,2)=B(2)
3380      Zoc_cal(K1,3)=B(3)
3390 !
3400    NEXT K
3410  SUBEND
3420  !-------------------------------------------------------------------------
3430  SUB Raw_red_dat(Buff,Zoc,@Data_path2)
3440 ! Routine to reduce raw data using polynomial:
3450    COM /Zoc_dat/ REAL Zoc_cal(96,10) BUFFER
3460    REAL Red_data(32) BUFFER
3470    REAL Raw_data(32)
3480    COM /Stats/ REAL Period,Sample_number
3490    FOR Block=1 TO Sample_number          ! Loop routine to transfer data
3500      ASSIGN @Buffer_path2 TO BUFFER Red_data(*);FORMAT OFF
3510      Input_rblock(Buff,Raw_data(*),32,1) ! Load data samples
3520      Sample_time=Block*Period
3530 !
3540 !   Red_data = a0 + a1*x + a2*x^2 + a3*x^3
3550 !
3560 ! where a0,a1,a2, & a3 are Least Square coefficients, and x is
3570 ! the individual port raw data value.
3580 !
3590      FOR K=1 TO 32
3600        K1=(Zoc-1)*32+K
3610        Red_data(K)=Zoc_cal(K1,0)+Zoc_cal(K1,1)*Raw_data(K)+Zoc_cal(K1,2)*Raw_data(K)^2+Zoc_cal(K1,3)*Raw_da
3620      NEXT K
3630 !
3640      Red_data(0)=Sample_time           ! Store reduce data sample time.
3650 !
3660      CONTROL @Buffer_path2,4;(32+1)*8  ! Close buffer when full
3670      TRANSFER @Buffer_path2 TO @Data_path2  ! Transfer data to hard drive
3680      ASSIGN @Buffer_path2 TO *
3690    NEXT Block
3700  SUBEND
3710 !-------------------------------------------------------------------------
```

Figure B3 (cont) Development Program: SCAN_ZOC_03

```
10   ! Program: SCAN_ZOC_04
20   ! Description: Application program to operate HP6944A collecting pressure
30   !              readings from 1-3 ZOC-14 32 port modules using the CAL2000
40   !              to provide calibration data, reduce raw pressure data and
50   !              store data to the hard drive.
60   ! Hardware:  (1) HP6944A Multi-processors
70   !             - (2) 500 kHz A/D Cards (HP69759A)
80   !             - (2) High Speed Memory Cards (HP69791A)
90   !             - (1) Timer/Pacer Card (HP69736A)
100  !             - (1) Counter Card (HP69775A)
110  !            (1) HiScan CAL2000 Calibration Module
120  !            (1) ZOC-14 32 port Electronic Pressure Scanning Modules
130  ! Notes: 1. This program utilizes up to three (3) Zoc Modules storing data
140  !        of each Zoc into a seperate buffer Memory System (HP69791A).
150  !        2. COM /Names/ line and BDAT file ZOC_CONFIG_04 must match for
160  !        this program to operate.
170  !        3. HiScan requires a short period to stabilize before reading
180  !        the pressure valves.  The Wait_for statement (line 340) sets this
190  !        wait period in seconds.  Adjustment of the variable my be required
200  !        as additional Zocs are integrated into the Data Acquisitior System.
210  !        4. HiScan currently configured for one (1) calibrator.  This
220  !        program is written to operate one (1) or two (2) calibrators.
230  !
240  ! Buffer Memory: 65536 16-bit data words in HP69791A per system
250  ! Timer: Maximum 32676 counts for one HP69775A
260  ! Max speed of HP system is Period=0.000002 sec. or 500 kHz.
270  !
280   COM /Isssсom/ INTEGER X(1:1106)
290   COM /.sss_heap/ Isss_heap(1000)
300   COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Timer
310  !Configure("Menu_off","ZOC_CONFIG_04")
320   Configure("Ask_me","ZOC_CONFIG_04")
330  !
340   Wait_for=1.5                        ! Wait time for HiScan stabilization
350  !
360 Input: !----- INPUT VARIABLES ---------------------------------------------
370   PRINT "Program:  SCAN_ZOC_04"
380   PRINT "  - Scans 1-3 Zoc-14 Modules (32 pressure sensing ports each)."
390   PRINT "  - Uses Zero Operate Calibrate (ZOC) principal:"
400   PRINT "      - Collects raw pressure data (Zero Operate)"
410   PRINT "      - Collects calibration data (Calibrate)"
420   PRINT "      - Reduces raw data and stores data on hard drive :,700,0,1"
430   PRINT "  - HiScan Calibration Module used for the reference pressure standard."
440   PRINT "  - Raw pressure data reduced using calibration data from HiScan"
450   PRINT "    and Zocs in the calibration mode."
460   PRINT "  - Program designed to operate up to three (3) Memory-A/D Cards"
470   PRINT "    simultaneously.  COM /Names/ line must match Multi-programmer"
480   PRINT "    (HP6944A) configuration.  ZOC_CONFIG_04 file must be updated"
490   PRINT "    to the revised COM /Names/ line when altered."
500   PRINT
510   PRINT "Input variables: Scan frequency (1-100,000 Hz)"
520   PRINT "                 Samples per Port (1-1021)"
530   PRINT "                 Number of Zocs and their capacity"
540   PRINT "                 Calibration/Reduced data file name ID"
550  !
560  ! COM assigns calibration data array for 32 Zoc ports and standard values.
570   COM /Zoc_dat/ REAL Zoc_cal1(33,10) BUFFER,Zoc_cal2(33,10) BUFFER,Zoc_cal3(33,10) BUFFER
580   COM /Stats/ REAL Period,Sample_number,Wait_for,INTEGER Cal_mod_id(3)
590   MAT Zoc_cal1= (0)
600   MAT Zoc_cal2= (0)
610   MAT Zoc_cal3= (0)
620   PRINT
630   INPUT "Enter data rate (1-100kHz):",Hz
640   INPUT "Number of samples per port (1-1021): ",Sample_number
650   INPUT "Number of Zoc's connected to Multi-programer",Zoc_number
```

Figure B4 Development Program: SCAN_ZOC_04

```
660   Cal_mod_id(0)=Zoc_number
670   FOR Zoc_case=1 TO Zoc_number
680     SELECT Zoc_case
690     CASE 1
700       INPUT "Enter Calibration Module number set for Zoc #1 (Enter 1 or 2):",Cal_mod_id(1)
710     CASE 2
720       INPUT "Enter Calibration Module number set for Zoc #2 (Enter 1 or 2):",Cal_mod_id(2)
730     CASE 3
740       INPUT "Enter Calibration Module number set for Zoc #3 (Enter 1 or 2):",Cal_mod_id(3)
750     END SELECT
760   NEXT Zoc_case
770   INPUT "Enter Calibration and Reduced Data file name ID (10 char max):",File_id$
780   Period=1/Hz
790   PRINT
800   PRINT "Data acquisition rate:";TAB(50);Hz;" Hz"
810   PRINT "Number of samples per port:";TAB(50);Sample_number
820   PRINT "Number of Zocs to be scanned:";TAB(50);Zoc_number
830   PRINT "Total number of ports to be scanned:";TAB(50);Zoc_number*32
840   PRINT "Total raw data acquisition time:";TAB(50);Period*Sample_number*32;" sec."
850   PRINT "Total calibration data acquisition time:";TAB(50);Period*5*32+(7*Wait_for);" sec."
860   !
870   Zoc_cal1(0,0)=Period
880   Zoc_cal1(0,1)=Sample_number
890   Zoc_cal1(0,2)=1
900   Zoc_cal1(0,3)=Cal_mod_id(1)
910   Zoc_cal2(0,0)=Period
920   Zoc_cal2(0,1)=Sample_number
930   Zoc_cal2(0,2)=2
940   Zoc_cal2(0,3)=Cal_mod_id(2)
950   Zoc_cal3(0,0)=Period
960   Zoc_cal3(0,1)=Sample_number
970   Zoc_cal3(0,2)=3
980   Zoc_cal3(0,3)=Cal_mod_id(3)
990   !
1000  PRINT
1010  PRINT "Ensure HiScan is on-line, calibration pressure source at 90 psi,"
1020  PRINT "and calibrator pressure cut-off valve is open (on back of HiScan)"
1030  !
1040  !
1050  CONTROL 9,5;3                    ! Set DTR & RTS to active for HiScan
1060  OUTPUT 9;VALS(1);"IC";CHR$(13);END! Initialize Calibrator module #1
1070  OUTPUT 9;VALS(2);"IC";CHR$(13);END! Initialize Calibrator module #2
1080  WAIT Wait_for                    ! Allow HiScan to set Zocs
1090  !
1100  DISP "Press F2 to start data aquisition"
1110  PAUSE
1120 !
1130 Collect_raw_dat: !--- COLLECT RAW PRESSURE DATA ---------------------------
1140  CLEAR SCREEN
1150  PRINT
1160  PRINT "Collecting raw pressure data."
1170  Count=Sample_number*32           ! Set Count as function of sample number
1180                                   ! and number of port readings (32) on
1190                                   ! Zoc for raw data collection.
1200  CALL Scan_zocs(Count,Period)     ! Collect raw data into Memory System
1210  PRINT
1220  PRINT "Raw data collection complete."
1230  BEEP
1240 !
1250 Raw_data_xfer:!------ TRANSFER RAW DATA FM MEMORY SYSTEM TO HARD DISC -----
1260  PRINT
1270 !
1280  FOR Zoc_case=1 TO Zoc_number     ! Collect raw data, reduce data and
1290    SELECT Zoc_case                ! and store reduce data on hard drive
1300    CASE 1
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
1310      CALL Raw_dat(Buffer1,1,Sample_number)
1320    CASE 2
1330      CALL Raw_dat(Buffer2,2,Sample_number)
1340    CASE 3
1350      CALL Raw_dat(Buffer3,3,Sample_number)
1360    END SELECT
1370  NEXT Zoc_case
1380  !
1390 Initial_cal:!-------- CALIBRATION SET-UP ---------------------------------
1400  ! Calibration data array for each Zoc: Zoc_cal_(33 10)
1410  ! Format:
1420  !   For ports i=1 to 33
1430  !       Row 0, column 0: Period
1440  !       Row 0, column 1: Sample number
1450  !       Row 0, column 2: Zoc #
1460  !       Row 0, column 3: Calibrator module ID (1=50 psi 2=15 psi)
1470  !       Row 0:  __ __ __ __ NH NM NL ZO PL PM PH  (pressure Hg.)
1480  !       Row i: A0 A1 A2 A3 NH NM NL ZO PL PM PH (LS coef,press volts)
1490  !    LS coef are Least Squares curve fit coef for third order polynomial.
1500  !
1510  PRINT
1520  PRINT "Collecting calibration data."
1530  REAL Cal1(1120),Cal2(1120),Cal3(1120)! Calibration data array
1540  Count=32*5                          ! Set count to collect calibration data
1550  DIM Command_mode$(1:7)[2]
1560  Command_mode$(1)="NH"
1570  Command_mode$(2)="NM"
1580  Command_mode$(3)="NL"
1590  Command_mode$(4)="ZO"
1600  Command_mode$(5)="PL"
1610  Command_mode$(6)="PM"
1620  Command_mode$(7)="PH"
1630  !
1640 Collect_cal_dat:!---- COLLECT RAW CALIBRATION DATA ------------------------
1650  !
1660  ! Collect raw calibration data for each HiScan setting
1670  FOR Index=1 TO 7
1680    CALL Cal2000(Command_mode$(Index),Index)
1690    CALL Scan_zocs(Count,Period)
1700    FOR Zoc_case=1 TO Zoc_number
1710      SELECT Zoc_case
1720      CASE 1
1730        Input_rblock(Buffer1,Cal1(*),160,(Index-1)*160+1)
1740      CASE 2
1750        Input_rblock(Buffer2,Cal2(*),160,(Index-1)*160+1)
1760      CASE 3
1770        Input_rblock(Buffer3,Cal3(*),160,(Index-1)*160+1)
1780      END SELECT
1790    NEXT Zoc_case
1800  NEXT Index
1810  !
1820  PRINT
1830  PRINT "Calibration data collection complete."
1840  BEEP
1850  WAIT .25
1860  BEEP
1870  OUTPUT 9;VAL$(1);"IC";CHR$(13);END! Initialize Calibrator module #1
1880  OUTPUT 9;VAL$(2);"IC";CHR$(13);END! Initialize Calibrator module #2
1890  PRINT
1900  PRINT "*** Secure Calibrator pressure valve to conserve Nitrogen ***"
1910  PRINT
1920  PRINT "HiScan Calibration modes and pressures (in Hg):"
1930 Fmt1:IMAGE /,5X,K,10X,K,10X,K,10X,K
1940  PRINT USING Fmt1;"Mode","Zoc #1","Zoc #2","Zoc #3"
1950 Fmt2:IMAGE 6X,K,10X,3D.4D,8X,3D.4D,8X,3D.4D
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
1960  FOR I=4 TO 10
1970    PRINT USING Fmt2;Command_mode$(I-3),Zoc_cal1(0,I),Zoc_cal2(0,I),Zoc_cal3(0,I)
1980  NEXT I
1990  !
2000 Reduce_cal_dat:!----- REDUCE CALIBRATION DATA AND STORE ON HARD DRIVE -----
2010 ! Routine to reduce Cal_(*) into Zoc_cal_(I,J) and store on hard drive
2020 !
2030  PRINT
2040  PRINT
2050  PRINT "Hard drive name =>  :,700,0,1"
2060  FOR Zoc_case=1 TO Zoc_number
2070    SELECT Zoc_case
2080    CASE 1
2090      CALL Zoc_dat_red(Cal1(*),Zoc_cal1(*),File_id$)
2100    CASE 2
2110      CALL Zoc_dat_red(Cal2(*),Zoc_cal2(*),File_id$)
2120    CASE 3
2130      CALL Zoc_dat_red(Cal3(*),Zoc_cal3(*),File_id$)
2140    END SELECT
2150  NEXT Zoc_case
2160 !
2170 Reduce_data:!----- REDUCE DATA AND STORE ON HARD DRIVE ------------------
2180 ! Routine loads raw and calibration data from hard drive, reduces the raw
2190 ! data, and stores the reduced data to the hard drive.
2200 !
2210  FOR Zoc_case=1 TO Zoc_number
2220    SELECT Zoc_case
2230    CASE 1
2240      CALL Raw_red_dat(Zoc_cal1(*),File_id$)
2250    CASE 2
2260      CALL Raw_red_dat(Zoc_cal2(*),File_id$)
2270    CASE 3
2280      CALL Raw_red_dat(Zoc_cal3(*),File_id$)
2290    END SELECT
2300  NEXT Zoc_case
2310    !
2320  BEEP
2330  WAIT .25
2340  BEEP
2350  WAIT .25
2360  BEEP
2370  !
2380 Finish:!
2390  PRINT
2400  PRINT "Available Memory: ";SYSTEM$("AVAILABLE MEMORY")
2410  DISP "Press F2 to continue and return to ZOC Menu."
2420  PAUSE
2430  LOAD "ZOC_MENU",10
2440 !
2450  END
2460 End:!=============================================================
2470 ! Routine to operate HP6944A to collect pressure data and store in Memory
2480 !Note: -Zoc address driven by Counter Card binary output at edge connection
2490 !       through a 7404 open collector DTL/TTL located in an auxiliary box.
2500 !
2510  SUB Scan_zocs(Count,Period)
2520    COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Timer
2530    Wait_time=INT(Count*Period)+10  ! Set Timer wait time to +10 secs.
2540    Init(Timer)                     ! Initialize Timer system
2550    Set_timeout(Timer,Wait_time)    ! Set Wait_for period of xx secs.
2560    Set_count(Timer,Count)          ! Set Count number into Timer
2570    Set_period(Timer,Period)        ! Set Timer pulse length in secs.
2580    Init(Buffer1)                   ! Initialize Buffer for data storage
2590    Init(Buffer2)
2600    !INIT(BUFFER3)
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
2610    Start(Timer)                    ! Start data sample collection
2620    Wait_for(Timer)                 ! Data samples stored in Memory System
2630  SUBEND
2640 !-------------------------------------------------------------------------
2650 ! Subroutine to collect raw pressure data from Memory System and store
2660 ! onto the hard drive for future data reduction.
2670  SUB Raw_dat(Buff,Zn,Sn)
2680    ON ERROR GOSUB Purge_file
2690    INTEGER Raw_data(1:32672) BUFFER ! Integer raw data buffer for 32*1021
2700                                     ! data samples.  Integer format for
2710                                     ! mininum transfer time to storage.
2720    Data_file$="ZRAW"&VAL$(Zn)       ! Raw data file
2730    Data_disc$=Data_file$&":,700,0,1"
2740    CREATE BDAT Data_disc$,1,2       ! Create BDAT file of 2 byte records.
2750    ASSIGN @Data_path TO Data_disc$ ! Assign path to hard drive
2760    ASSIGN @Buffer_path TO BUFFER Raw_data(*);FORMAT OFF
2770    Input_iblock(Buff,Raw_data(*),Sn*32,1)      ! Load data samples
2780    CONTROL @Buffer_path,4;32*2*Sn  ! Close buffer when full
2790    TRANSFER @Buffer_path TO @Data_path     ! Transfer data to hard drive
2800    ASSIGN @Buffer_path TO *
2810    ASSIGN @Data_path TO *
2820    PRINT "Raw pressure data for Zoc #";Zn;" transferred to the hard drive file ";Data_file$
2830    GOTO Fin
2840 Purge_file: !
2850    IF ERRN=54 THEN
2860      PURGE Data_disc$
2870    END IF
2880    RETURN
2890 Fin:     !
2900  SUBEND
2910 !-------------------------------------------------------------------------
2920 ! Subroutine controls calibration mode and reads pressure from Pressure
2930 ! Standard into Zoc_cal(*) array.
2940  SUB Cal2000(Command$,I)
2950    COM /Zoc_dat/ REAL Zoc_cal1(*) BUFFER,Zoc_cal2(*) BUFFER,Zoc_cal3(*) BUFFER
2960    COM /Stats/ REAL Period,Sample_number,Wait_for,INTEGER Cal_mod_id(3)
2970    DIM Pressure$[5]                        ! Required to read data stream
2980    OUTPUT 9;VAL$(1);Command$;CHR$(13);END  ! Sets calibrator #1 mode
2990    OUTPUT 9;VAL$(2);Command$;CHR$(13);END  ! Sets calibrator #2 mode
3000    WAIT Wait_for                           ! Allow HiScan to stabilize
3010    FOR K=1 TO Cal_mod_id(0)                ! Read HiScan cal press
3020      SELECT K
3030      CASE 1
3040        OUTPUT 9;VAL$(Cal_mod_id(1));"RP";CHR$(13);END
3050        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal1(0,I+3),Pressure$
3060      CASE 2
3070        OUTPUT 9;VAL$(Cal_mod_id(2));"RP";CHR$(13);END
3080        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal2(0,I+3),Pressure$
3090      CASE 3
3100        OUTPUT 9;VAL$(Cal_mod_id(3));"RP";CHR$(13);END
3110        ENTER 9 USING "#,SD.5DESZZ,K";Zoc_cal3(0,I+3),Pressure$
3120      END SELECT
3130    NEXT K
3140    IF I<=3 THEN                        ! Account for positive pressures used
3150      Zoc_cal1(0,I+3)=-Zoc_cal1(0,I+3) ! by HiScan in the NH,NM, & NL mode.
3160      Zoc_cal2(0,I+3)=-Zoc_cal2(0,I+3)
3170      Zoc_cal3(0,I+3)=-Zoc_cal3(0,I+3)
3180    END IF
3190  SUBEND
3200 !-------------------------------------------------------------------------
3210 ! Subroutine reduces calibration data collected from Memory System and
3220 ! HiScan calibration pressure data into a Third-order polynomial curve
3230 ! fit using the Least Squares routine.  The complete calibration array
3240 ! Zoc_cal_ is then stored onto the hard drive.
3250  SUB Zoc_dat_red(REAL Cal(*),Zoc_cal(*) BUFFER,F_id$)
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
3260    REAL A(3,3),B(3),C(3),Sum_x(6),A_inv(3,3)! Least Square reduction arrays
3270 !
3280 ! Converting Cal(*) to Zoc_cal(*)
3290    FOR J=4 TO 10                    ! Cal runs: NH,NM,NL,ZO,PL,PM,PH
3300      FOR I=1 TO 32                  ! Zoc ports per calibration run
3310        FOR K=0 TO 4                 ! Number of samples per run
3320          Zoc_cal(I,J)=Zoc_cal(I,J)+Cal(I+K*32+(J-4)*160)
3330        NEXT K
3340        Zoc_cal(I,J)=Zoc_cal(I,J)/5 ! Average of 5 samples per port I
3350      NEXT I
3360    NEXT J
3370 !
3380 ! Calibration data reduction using Least Squares Polynominal fitting.
3390    FOR K=1 TO 32                ! Loop for each port
3400 !
3410      MAT C= (0)
3420      MAT Sum_x= (0)
3430 !
3440      FOR J=1 TO 6               ! Routine to reduce individual port cal
3450        FOR I=4 TO 10            ! data into elements to a power x^j
3460          Sum_x(J)=Sum_x(J)+Zoc_cal(K,I)^J
3470        NEXT I
3480      NEXT J
3490 !
3500      FOR I=0 TO 3               ! Derive A array
3510        FOR J=0 TO 3
3520          A(I,J)=Sum_x(I+J)
3530        NEXT J
3540      NEXT I
3550      A(0,0)=7
3560 !
3570      FOR J=0 TO 3               ! Derive C array
3580        FOR I=4 TO 10
3590          C(J)=C(J)+Zoc_cal(K,I)^J*Zoc_cal(0,I)
3600        NEXT I
3610      NEXT J
3620 !
3630      MAT A_inv= INV(A)
3640      MAT B= A_inv*C             ! B array is matrix of Least Square
3650 !                                 coefficients a0,a1,a2,& a3 for polynomial
3660 !                                 equation fitting calibration data for a
3670 !                                 specified port
3680 !
3690 ! Collect Least Square coefficients
3700      Zoc_cal(K,0)=B(0)
3710      Zoc_cal(K,1)=B(1)
3720      Zoc_cal(K,2)=B(2)
3730      Zoc_cal(K,3)=B(3)
3740 !
3750    NEXT K
3760 !
3770 ! Transfer calibration data to hard drive.
3780    ON ERROR GOSUB Purge_file
3790    DIM Data_disc$[23]              ! Define string for data file name
3800    Data_file$="ZC"&VAL$(Zoc_cal(0,2))&"_"&F_id$ ! Zoc calibration file name
3810    Data_disc$=Data_file$&":,700,0,1"
3820    CREATE BDAT Data_disc$,33,8*11  ! Create BDAT file of 11*8 byte
3830    ASSIGN @Data_path TO Data_disc$ ! Assign path to hard drive
3840    ASSIGN @Buffer_path TO BUFFER Zoc_cal(*);FORMAT OFF
3850    CONTROL @Buffer_path,4;8*11*33  !Set data file length
3860    TRANSFER @Buffer_path TO @Data_path!Store cal data on hard drive
3870    ASSIGN @Buffer_path TO *        ! Close path
3880    ASSIGN @Data_path TO *          ! Close path
3890    PRINT "Calibration data for Zoc #";Zoc_cal(0,2);" transfer to the hard drive file ";Data_file$
3900    GOTO Fin
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
3910 Purge_file:   !
3920    IF ERRN=54 THEN
3930      PURGE Data_disc$
3940    END IF
3950    RETURN
3960 Fin:              !
3970  SUBEND
3980  !-------------------------------------------------------------------------
3990  ! Subroutine loads raw data from the hard drive, reduces the data using
4000  ! calibration coeficients, and stores the reduced data onto the hard drive
4010  SUB Raw_red_dat(REAL Zoc_cal(*) BUFFER,F_id$)
4020    COM /Names/ Buffer1,Adc1,Buffer2,Adc2,Timer
4030    INTEGER Data_integer(1:32) BUFFER
4040    REAL Data_real(1:32),Data(32) BUFFER
4050    DIM Data_file2$[23]
4060    ON ERROR GOSUB Purge_file
4070    Data_file1$="ZRAW"&VAL$(Zoc_cal(0,2))
4080    Data_disc1$=Data_file1$&":,700,0,1"
4090    Data_file2$="ZR"&VAL$(Zoc_cal(0,2))&"_"&F_id$
4100    Data_disc2$=Data_file2$&":,700,0,1"
4110    CREATE BDAT Data_disc2$,1,8*33  ! Create BDAT file of 33*8 byte records.
4120    ASSIGN @Data_path1 TO Data_disc1$
4130    ASSIGN @Data_path2 TO Data_disc2$
4140    CONTROL @Data_path1,5;2            !Set read pointer to first data byte
4150    !
4160    ! Recover raw data, convert to real, reduce then store in blocks
4170    ! of samples (32 ports scanned per block)
4180    FOR Block=1 TO Zoc_cal(0,1)
4190      ASSIGN @Buffer_path1 TO BUFFER Data_integer(*);FORMAT OFF
4200      TRANSFER @Data_path1 TO @Buffer_path1;COUNT 32*2
4210      CONTROL @Buffer_path1,4;32*2
4220      SELECT Zoc_cal(0,2)
4230      CASE 1
4240        Translate(Adc1,Data_integer(*),Data_real(*))
4250      CASE 2
4260        Translate(Adc2,Data_integer(*),Data_real(*))
4270      CASE 3
4280      END SELECT
4290    !
4300 ! Routine to reduce raw data using polynomial:
4310 !
4320 !       Data = a0 + a1*x + a2*x^2 + a3*x^3
4330 !
4340 ! where a0,a1,a2, & a3 are Least Square coefficients, and x is
4350 ! the individual port raw data value.
4360 !
4370      Sample_time=Zoc_cal(0,0)*(Block-1)
4380      Data(0)=Sample_time              ! Store reduce data sample time.
4390      FOR K=1 TO 32
4400        Data(K)=Zoc_cal(K,0)+Zoc_cal(K,1)*Data_real(K)+Zoc_cal(K,2)*Data_real(K)^2+Zoc_cal(K,3)*Data_real(K)^
4410      NEXT K
4420 !
4430      ASSIGN @Buffer_path2 TO BUFFER Data(*)
4440      CONTROL @Buffer_path2,4;8*33
4450      TRANSFER @Buffer_path2 TO @Data_path2
4460      ASSIGN @Buffer_path2 TO *
4470    NEXT Block
4480    !
4490    ASSIGN @Data_path1 TO *
4500    ASSIGN @Data_path2 TO *
4510    ASSIGN @Buffer_path1 TO *
4520    PRINT "Reduced data file for Zoc #";Zoc_cal(0,2);" transfer to the hard drive file ";Data_file2$
4530    GOTO Fin
4540 Purge_file:   !
4550    IF ERRN=54 THEN
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

```
4560       PURGE Data_disc2$
4570     END IF
4580     RETURN
4590 Fin:    !
4600  SUBEND
4610 !-------------------------------------------------------------------------
```

Figure B4 (cont) Development Program: SCAN_ZOC_04

APPENDIX C. DATA FILE MANAGEMENT

SCAN_ZOC_05 generates a large number of data files in a short period of time. During program development, it was desirable to purge these files from the "DATA" hard drive ":,700,0,1" to keep track of useable files and remove unwanted files. The utility program "PURGE_PROG", Figure C1, was developed to purge SCAN_ZOC_05-generated data files from the hard drive. This program can be used when known bad data has been collected and it is desirable to purge these files from the hard drive.

PURGE_PROG is located on the UTILity directory (Figure D1). It is loaded into RAM from any directory and run by typing,

1. LOAD "/UTIL/PURGE_PROG", followed by depressing the <Enter> key to load the program into the RAM.
2. RUN, followed by the <Enter> key to execute the program.

The program executes a CAT command and displays the hard drive files as illustrated in Figure C2. The program prompts the user for the date (YMMDD), the first run number, and last run number of consecutive data files to be purged. Program actions are displayed on the screen. The program prompts the user to purge more files or to re-display the remaining files

using the CAT command. If file re-display (2=CAT) is selected, the program prompts again for the date and run numbers. Entering <0,0,0> will exit the program.

```
10    !Program to purge designated files from SCAN_ZOC_05
20    CLEAR SCREEN
30    ON ERROR GOTO Cont
40    Drive$=":,700,0,1"
50 Input1:   !
60    CAT Drive$;SELECT "Z"
70    PRINT
80 Input2:  !
90    INPUT "Enter date (YMMDD), begin run#, end run# (0,0,0=Exit)",Date$,Run0,Run1
100   IF Run0=0 THEN Fin
101   FOR Zoc=1 TO 3
102     FOR Run=Run0 TO Run1
110       File1$="ZW"&VAL$(Zoc)&Date$&VAL$(Run)
120       File2$="ZC"&VAL$(Zoc)&Date$&VAL$(Run)
130       File3$="ZR"&VAL$(Zoc)&Date$&VAL$(Run)
140 Purge1:  !
150       PURGE File1$&Drive$
160       PRINT "Purged file: ";File1$
170 Purge2:  !
180       PURGE File2$&Drive$
190       PRINT "Purged file: ";File2$
200 Purge3:  !
210       PURGE File3$&Drive$
220       PRINT "Purged file: ";File3$
280       GOTO Next
290 Cont:  !
300       IF ERRN=56 THEN
310         SELECT ERRLN
320         CASE 150
330           GOTO Purge2
340         CASE 180
350           GOTO Purge3
360         CASE 210
370           GOTO Next
380         END SELECT
390       END IF
400       RETURN
401 Next:  !
402     NEXT Run
403   NEXT Zoc
405 Print:  !
406   PRINT
407   INPUT "Purge more? (0=No 1=Yes 2=CAT)",Act
408   IF Act=1 THEN Input2
409   IF Act=2 THEN Input1
410 Fin:  !
420   LOAD "ZOC_MENU",10
430   END
```

Figure C1 TPL Program: PURGE_PROG

```
:,700,0,1
VOLUME LABEL: DATA
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE      TIME

ZW1205161     BDAT        97        2       592   16-May-92 14:54
ZW2205161     BDAT        97        2       594   16-May-92 14:54
ZW3205161     BDAT        97        2       596   16-May-92 14:54
ZC1205161     BDAT        33       88       598   16-May-92 14:55
ZC2205161     BDAT        33       88       611   16-May-92 14:55
ZC3205161     BDAT        33       88       624   16-May-92 14:55
ZR1205161     BDAT         3      264       637   16-May-92 14:55
ZR2205161     BDAT         3      264       642   16-May-92 14:55
ZR3205161     BDAT         3      264       647   16-May-92 14:55
```

Figure C2 Hard Drive Sample Listing of ZOC-14 DAS Data Files

APPENDIX D. TPL PROGRAMS

A selection of programs generated for use at TPL are documented in this Appendix. The programs are found in individual directories according to their specific function or purpose. Figure D1 displays the major directories and the associated programs. Each directory is identified by the forward slash character "/", followed by the directory name, and a colon, followed by the system drive name "CS80:,700". Subdirectories are identified by the FILE TYPE "DIR". BASIC programs are identified by the FILE TYPE "PROG".

"AUTOST", Figure D2, is the HP9000 initialization program located in directory "/WORKSTATIONS". This program is called by the BASIC system during boot-up. AUTOST is used to display the Main Menu and define the function keys used to select the Sub Menus and associated programs in the HP9000 Data Acquisition System.

Turbocharger performance mapping programs (prepared for student laboratories) are listed in Figures D3-D6. The program "SCAN_TEMP", Figure D7, is used to monitor temperature probes. SCAN_TEMP, as listed, monitors temperature probe outputs from the Turbocharger Test Cell. "TURBO_MENU", Figure D8, is used to select the turbocharger the above programs (Figures D3-D7).

Turbomachinery design programs used in the course AE 4431 are listed in Figures D9-D13.

Program "SCAN", Figure D14, is used to operate the HG-78K Scanivalve Controller, HP3495A Scanner, HP3456A Digital Voltmeter, and records Scanivalve transducers. SCAN and SCAN_TEMP programs, located in directory /WORK/DEVICE_PROG, are used in the majority of programs required for data acquisition at TPL.

The sub-program "Plot", Figure D15, is appended to the end of programs used to display plots on the HP9000 CRT, or for generating plots on the various plotters.

The sub-program "FNDate$", Figure D16, is a user-defined function to calculate numbered values for the date in the format year, month, and day. FNDate$ is in sub-program "Date_func", which is located in the directory /WORK/FUNCTION_PROG.

"FILE_XFER", Figure D17, is a utility program used for file transfer and "purging" on the HP9000. FILE_XFER is located in the /UTIL directory.

The program "MAIN_MENU", Figure D18, is used to quickly restore the CRT screen to its system display and call-up the Main Menu screen. The program is located in the root directory /. Loading and executing MAIN_MENU is accomplished by entering the following command on the HP9000, and executing the command by depressing the Enter key:

LOAD "/MAIN_MENU",10 <Enter>

/WORKSTATIONS:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE: 23328

FILE NAME	FILE TYPE	NUM RECS	REC LEN	MODIFIED DATE	TIME	PERMISSION	OWNER	GROUP
BIN5.1	DIR	21	32	17-Jan-92	14:25	RWXRWXRWX	18	9
STUDENT_DIARY	DIR	87	32	14-Apr-92	17:41	RWXRWXRWX	18	9
AUTOST	PROG	11	256	29-Apr-92	15:32	RW-RW-RW-	18	9
DIRECTORY_INFO	PROG	34	256	30-Apr-92	15:11	RW-RW-RW-	18	9

/TURBINE:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE: 23328

FILE NAME	FILE TYPE	NUM RECS	REC LEN	MODIFIED DATE	TIME	PERMISSION	OWNER	GROUP
TURBO2	PROG	14	256	1-Oct-91	17:40	RW-RW-RW-	18	9
TURBO3	PROG	14	256	1-Oct-91	17:43	RW-RW-RW-	18	9
TURBO4	PROG	34	256	16-Oct-91	16:01	RW-RW-RW-	18	9
TURBO1	PROG	45	256	21-Oct-91	7:59	RW-RW-RW-	18	9
TURBO_MENU	PROG	7	256	29-Apr-92	15:52	RW-RW-RW-	18	9
SCAN_TEMP	PROG	8	256	30-Apr-92	7:55	RW-RW-RW-	18	9
TURBO_BAK	DIR	4	32	20-Apr-92	13:49	RWXRWXRWX	18	9

/HP6944A:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE: 23528

FILE NAME	FILE TYPE	NUM RECS	REC LEN	MODIFIED DATE	TIME	PERMISSION	OWNER	GROUP
fp_data	BDAT	18	256	3-Jan-92	7:52	RW-RW-RW-	18	9
CARD_TEST	PROG	444	256	3-Jan-92	7:52	RW-RW-RW-	18	9
CONFIGURE	PROG	127	256	3-Jan-92	7:53	RW-RW-RW-	18	9
LIBRARY_5	PROG	888	256	3-Jan-92	7:54	RW-RW-RW-	18	9
VERIFY	PROG	282	256	3-Jan-92	8:02	RW-RW-RW-	18	9
menu_data	BDAT	71	256	3-Jan-92	12:20	RW-RW-RW-	18	9
MENUER	PROG	136	256	3-Jan-92	13:40	RW-RW-RW-	18	9
FRONT_P	PROG	174	256	3-Jan-92	13:41	RW-RW-RW-	18	9
HP6944A_BAK	DIR	8	32	30-Apr-92	12:57	RWXRWXRWX	18	9
TABULATE_ZOC	PROG	13	256	26-Apr-92	15:40	RW-RW-RW-	18	9
READ_ZOC	PROG	11	256	26-Apr-92	17:30	RW-RW-RW-	18	9
SCAN_ZOC_05	PROG	1148	256	30-Apr-92	13:02	RW-RW-RW-	18	9
CAL_READ_PR1	PROG	12	256	30-Apr-92	13:49	RW-RW-RW-	18	9
LS_PLOT	PROG	22	256	30-Apr-92	14:33	RW-RW-RW-	18	9
DOCUMENT	PROG	59	256	13-Apr-92	8:45	RW-RW-RW-	18	9
VER_CFG	BDAT	6	256	24-Apr-92	16:51	RW-RW-RW-	18	9
HP6944A_MENU	PROG	7	256	7-Apr-92	7:18	RW-RW-RW-	18	9
DEVELOPMENT	DIR	11	32	30-Apr-92	15:07	RWXRWXRWX	18	9
PLOT_DATA	PROG	29	256	30-Apr-92	14:18	RW-RW-RW-	18	9
ZOC_CONFIG_05	BDAT	5	256	26-Apr-92	16:53	RW-RW-RW-	18	9
ZOC_MENU	PROG	9	256	24-Apr-92	12:55	RW-RW-RW-	18	9

Figure D1 Listing of TPL Programs by Directory

```
/DESIGN:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE:      23528
                  FILE     NUM    REC      MODIFIED
FILE NAME         TYPE    RECS    LEN DATE         TIME  PERMISSION OWNER GROUP
================ ===== ======= ===== ========= ===== ========== ===== =====
DESIGN_BAK        DIR         4     32 30-Apr-92  8:07  RWXRWXRWX     18     9
DESIGN_MENU       PROG        8    256 16-Dec-91 16:41  RW-RW-RW-     18     9
R_4431T           PROG       21    256 16-Dec-91 13:59  RW-RW-RW-     18     9
A_4431T           PROG       21    256 16-Dec-91 14:01  RW-RW-RW-     18     9
TURB3             PROG       29    256 17-Dec-91 14:32  RW-RW-RW-     18     9
TURB4             PROG       38    256 20-Dec-91 16:42  RW-RW-RW-     18     9


/WORK:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE:      23528
                  FILE     NUM    REC      MODIFIED
FILE NAME         TYPE    RECS    LEN DATE         TIME  PERMISSION OWNER GROUP
================ ===== ======= ===== ========= ===== ========== ===== =====
FUNCTION_PROG     DIR        12     32 20-Apr-92 14:35  RWXRWXRWX     18     9
DEVICE_PROG       DIR        22     32 30-Apr-92 15:29  RWXRWXRWX     18     9
DISCREP_DOC       PROG        5    256 30-Dec-91 11:59  RW-RW-RW-     18     9


/WORK/DEVICE_PROG:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE:      23528
                  FILE     NUM    REC      MODIFIED
FILE NAME         TYPE    RECS    LEN DATE         TIME  PERMISSION OWNER GROUP
================ ===== ======= ===== ========= ===== ========== ===== =====
CAL               PROG        7    256 20-Apr-92 14:15  RW-RW-RW-     18     9
CAL_READ_PR       PROG       11    256 20-Apr-92 14:16  RW-RW-RW-     18     9
SET_BRIDGE        PROG        6    256 20-Apr-92 14:16  RW-RW-RW-     18     9
READ_PORT         PROG        5    256 20-Apr-92 14:16  RW-RW-RW-     18     9
SCAN_PORT         PROG        4    256 20-Apr-92 14:17  RW-RW-RW-     18     9
BCD_CONVERT       PROG        4    256 20-Apr-92 14:17  RW-RW-RW-     18     9
SCAN_TEMP         PROG        8    256 20-Apr-92 14:18  RW-RW-RW-     18     9
UNPACK            PROG        4    256 20-Apr-92 14:23  RW-RW-RW-     18     9
HP37_PACK         PROG        5    256 20-Apr-92 14:23  RW-RW-RW-     18     9
HP37_BENCH        PROG        7    256 20-Apr-92 14:24  RW-RW-RW-     18     9
HP37_BCD          PROG        2    256 20-Apr-92 14:24  RW-RW-RW-     18     9
HP37-56           PROG        3    256 20-Apr-92 14:24  RW-RW-RW-     18     9
HP56_PACK         PROG        7    256 20-Apr-92 14:24  RW-RW-RW-     18     9
HP37_EXT1         PROG        4    256 20-Apr-92 14:24  RW-RW-RW-     18     9
HP56SRQ           PROG        3    256 20-Apr-92 14:26  RW-RW-RW-     18     9
HP56MEM           PROG        5    256 20-Apr-92 14:26  RW-RW-RW-     18     9
HP37_EXT2         PROG        4    256 20-Apr-92 14:26  RW-RW-RW-     18     9
HP37_A(*)         PROG        3    256 20-Apr-92 14:26  RW-RW-RW-     18     9
HP37_GRAPH        PROG       12    256 20-Apr-92 14:27  RW-RW-RW-     18     9
HP37_INT          PROG       12    256 20-Apr-92 14:27  RW-RW-RW-     18     9
SCAN              PROG       11    256 27-Apr-92 14:13  RW-RW-RW-     18     9
OUTPUT_9          PROG        5    256 30-Apr-92 15:29  RW-RW-RW-     18     9
```

Figure D1 (cont) Listing of TPL Programs by Directory

/WORK/FUNCTION_PROG:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE: 23528

FILE NAME	FILE TYPE	NUM RECS	REC LEN	MODIFIED DATE	TIME	PERMISSION	OWNER	GROUP
GRAPH	PROG	9	256	20-Apr-92	14:20	RW-RW-RW-	18	9
GRAPH_PLOTTER	PROG	6	256	20-Apr-92	14:20	RW-RW-RW-	18	9
Plot	PROG	10	256	20-Apr-92	14:20	RW-RW-RW-	18	9
Date_func	PROG	4	256	20-Apr-92	14:20	RW-RW-RW-	18	9
Square	PROG	4	256	20-Apr-92	14:21	RW-RW-RW-	18	9
BUFF1	PROG	9	256	20-Apr-92	14:21	RW-RW-RW-	18	9
LEAST_SQUARE	PROG	6	256	20-Apr-92	14:21	RW-RW-RW-	18	9
LS_TEST	PROG	18	256	20-Apr-92	14:21	RW-RW-RW-	18	9
LS_PLOT1	PROG	17	256	20-Apr-92	14:21	RW-RW-RW-	18	9
TEST	BDAT	4	256	20-Apr-92	14:22	RW-RW-RW-	18	9
DATA_LOGGIN	PROG	4	256	20-Apr-92	14:22	RW-RW-RW-	18	9
STOPWATCH	PROG	4	256	20-Apr-92	14:23	RW-RW-RW-	18	9

/UTIL:CS80, 700
LABEL:
FORMAT: HFS
AVAILABLE SPACE: 23528

FILE NAME	FILE TYPE	NUM RECS	REC LEN	MODIFIED DATE	TIME	PERMISSION	OWNER	GROUP
VERIFY	PROG	153	256	31-Jul-91	16:54	RW-RW-RW-	18	9
HPIB_CHECK	PROG	4	256	9-Aug-91	9:03	RW-RW-RW-	18	9
BACKUP	PROG	397	256	23-Aug-91	14:03	RW-RW-RW-	18	9
RETURN_MAIN	PROG	2	256	20-Apr-92	14:34	RW-RW-RW-	18	9
MAIN_MENU	PROG	2	256	20-Apr-92	16:02	RW-RW-RW-	18	9
PURGE_PROG	PROG	6	256	26-Apr-92	15:49	RW-RW-RW-	18	9
COPY_FILES	PROG	4	256	30-Apr-92	7:45	RW-RW-RW-	18	9
FILE_XFER	PROG	6	256	30-Apr-92	8:20	RW-RW-RW-	18	9
LASERJET	PROG	3	256	30-Apr-92	15:58	RW-RW-RW-	18	9

Figure D1 (cont) Listing of TPL Programs by Directory

```
10    !Program: AUTOST
20    !Description:  Program is loaded immediately following system boot-up
30    !              providing an initial User to Computer interface through
40    !              Function Key menu selection.
50    CLEAR SCREEN
60    KEY LABELS ON
70    DUMP DEVICE IS 711
80    ON KEY 1 LABEL " TURBO  CHARGER" GOTO Turbo
90    ON KEY 2 LABEL "COMPRESSLAB" GOTO Compressor
100   ON KEY 3 LABEL "DESIGN" GOTO Design
110   ON KEY 4 LABEL "MULTI   PROGRAM" GOTO Hp6944a
120   ON KEY 5 LABEL "WORK    DIR " GOTO Work
130   ON KEY 6 LABEL "BACKUP" GOTO Backup
140   ON KEY 7 LABEL "PROGRAM LISTINGS" GOTO Directory
150   ON KEY 8 LABEL "EXIT    MENU" GOTO Exit
160   !
170   PRINT "HP9000 Series 300 Computer Data Acquisition System"
180   PRINT
190   PRINT "Item:                              Select Function Key"
200   PRINT
210   PRINT "  Turbocharger Lab (*)                        F1"
220   PRINT "  Transonic Compressor Lab (*)                F2"
230   PRINT "  Turbine Design Programs (*)                 F3"
240   PRINT "  Multi-Programmer Opertion (HP6944A)         F4"
250   PRINT "  Work directory (*)                          F5"
260   PRINT "  Backup Files                                F6"
270   PRINT "  HP9000 Program Listing and Information      F7"
280   PRINT "  Exit Menu                                   F8"
290   PRINT
300   PRINT "Note: Binaries required for HP14753A CAT program are loaded"
310   PRINT "      to operate the HP6944A Multi-programmer."
320   PRINT "      Additional binaries are loaded with selection (*)."
330   PRINT
340   PRINT "      If Error 2 occurs: Memory overflow; reboot system using"
350   PRINT "      SYSBOOT statement and reselect desired directory."
360   !
370 Hold:  !
380   GOTO Hold
390 Turbo:  !
400   GOSUB Load_add_bin
410   MASS STORAGE IS "/TURBINE"
420   LOAD "TURBO_MENU",10
430 Compressor:          !
440   GOTO Hold
450 Design: !
460   GOSUB Load_add_bin
470   MASS STORAGE IS "/DESIGN"
480   LOAD "DESIGN_MENU",10
490 Hp6944a:  !
500   CLEAR SCREEN
510   MASS STORAGE IS "/HP6944A"
520   LOAD "HP6944A_MENU",10
530 Work:   !
540   GOSUB Load_add_bin
550   MASS STORAGE IS "/WORK"
560   GOTO Exit
570 Backup:  !
580   LOAD "/UTIL/BACKUP",10
590   GOTO Exit
600 Directory:!
610   MASS STORAGE IS "/WORKSTATIONS"
620   LOAD "DIRECTORY",10
630   GOTO Exit
640 Load_add_bin:!
650   LOAD BIN "/WORKSTATIONS/BIN5.1/GRAPHX"
```

Figure D2 TPL Program: AUTOST

```
660    LOAD BIN "/WORKSTATIONS/BIN5.1/TRANS"
670    LOAD BIN "/WORKSTATIONS/BIN5.1/XREF"
680    LOAD BIN "/WORKSTATIONS/BIN5.1/SERIAL"
690    LOAD BIN "/WORKSTATIONS/BIN5.1/COMPLEX"
700    RETURN
710 Exit:CLEAR SCREEN
720    END
```

Figure D2 (cont) TPL Program: AUTOST

```
10    !    FILE NAME:  "TURBO1"
20    !    DISK LABEL:  "/TURBINE"
30    !    DESCRIPTION:
40    !    THIS PROG RECORDS AND REDUCES
50    !    RAW DATA FOR THE T-18A40 TURBO-CHARGER
60    !    AS SET UP IN APRIL/MAY 1980 (RESET AUG 1991)
70    !    CHAN & PORT DESIGNATIONS ARE UNIQUE
80    !
90    !    VARIABLES FOR S/V SECTION
100   !      V=DESIRED S/V
110   !      P=PRESENT S/V PORT
120   !    VARIABLES FOR TEMP SECTION
130   !      SS=SCANNER LISTEN CODE
140   !      S=SCANNER #
150   !      C=CHANNEL
160   !      V=DVM READING
170   !    AUTHOR: TERRY EARGLE
180   !    DATE: JUL 79 (MOD MAY 80 BY M.J. KAISER)
190   !   MODIFIED BY MIKE JEDWAB 1987
200   !   MODIFIED BY KELLYHARRIS/ALAN MCGUIRE 1984
210   !   MODIFIED BY RICK WENDLAND AUG 1991
220   !
230   !   PROMPTS ADDED> ATMOSPHERIC PRESS.
231   !   RAW AND REDUCED DATA FILES USE DATE_RUN# CONTEXT IN THEIR NAME.
240   !   REDUCTION CHANGED TO USE A MEASURED STATIC PRESS FOR TURBINE
250   !   PRESSURE RATIO, I.E. INPUT STAGNATION PRESS, OUTPUT STATIC PRESS.
260   !
270   !   SCANIVALUE #1 SET TO ADVANCE ONLY!!!!!!!!!
280   CLEAR SCREEN
290   PRINTER IS CRT
300   DISP "PLEASE WAIT WHILE RESETTING SCANIVALVE"
310   V=1
320   A=1
330   GOSUB Read
340   PRINT "DATA RUN FOR A TURBO-CHARGER"
350   PRINT
360   DIM X(20),Y(6),Q(8),M(23),T(19),Z(20),A$[14],B$[14]
370   E1=1                              !Sets first record #=1
380   INPUT "STORING DATA ? (YES=1 NO=0)",G1
390   INPUT "ENTER MONTH,DAY,YEAR",X(3),X(5),X(7)
400   PRINT USING "K,DD,""/"",DD,""/"",DD";"DATE OF RUN: ",X(3),X(5),X(7)
410   INPUT "ENTER RUN #:",X(9)
420   IF G1=0 THEN 510
430   A$="RAW_"&VAL$(X(7))&VAL$(X(3))&VAL$(X(5))&"_"&VAL$(X(9))
440   B$="RED_"&VAL$(X(7))&VAL$(X(3))&VAL$(X(5))&"_"&VAL$(X(9))
450   CREATE ASCII A$&":,700,0,1",10    !Creates raw data file
460   CREATE ASCII B$&":,700,0,1",10    !Creates reduced data file
470   ASSIGN @Path1 TO A$&":,700,0,1"
480   ASSIGN @Path2 TO B$&":,700,0,1"
490   PRINT "RAW DATA FILE NAME:      ";A$
500   PRINT "REDUCED DATA FILE NAME:  ";B$
510   X(11)=1                           !Sets first datat point #=1
520   GOTO 600
530   GOSUB Save
540   FOR I=2 TO 20 STEP 2              !Reset X(*) for pressures to zero
550   X(I)=0
560   NEXT I
570   FOR I=1 TO 5                      !Reset Y(*) for temp & RPM to zero
580   Y(I)=0
590   NEXT I
600   PRINT USING "/,K,2D,4X,K,2D";"RUN #",X(9),"DATA PT #",X(11)
610   GOSUB 1130
620   !****** RECORD PRESSURES **********************************************
630   OUTPUT 722;"F1R1M0Z16STG1STIT3"!Sets-up HP3456A DVM
640   V=1                               !Scanivalve #1 designated
```

Figure D3 TPL Program: TURBO1

```
650   S=1                              !HP3495A Scanner #1 designated
660   PRINT USING "/,5X,K,DDD,/,2X,K,8X,K";"SCANIVALVE #",V,"PORT","IN. H2O"
670   FOR A=1 TO 10                    !Read scanivalve port values
680   GOSUB 970
690   WAIT 1.5
700   OUTPUT 701;"C"                   !Clears Scanner #1
710   OUTPUT 701 USING "DDD";V+9       !Scanner #1 set to read Scaniv port
720   TRIGGER 722                      !Trigger DVM
730   ENTER 722;V0                     !Read Scaniv port data from DVM
740   A2=A*2
750   X(A2)=V0*100000                  !Save Scaniv port reading in X(*)
760   PRINT USING "2X,DDD,7X,7D.D";P,X(A2)
770   NEXT A
780   OUTPUT 701;"C"                   !Clear scanner 1
790   GOSUB 1520
800   PRINT "-------------------------------------------------------------------------------"
810   DISP "PLEASE WAIT WHILE RESETTING SCANIVALVE"
820   A=1
830   V=1
840   GOSUB Read
850   INPUT "RETAKE SAME DATA POINT? (1=YES 0=NO)",U1
860   IF U1=1 THEN 540
870   IF G1=0 THEN 890
880   GOSUB Save
890   INPUT "COLLECT ANOTHER DATA POINT? (1=YES 0=NO)",R
900   X(11)=X(11)+1                    !Increment data point number
910   IF R=1 AND G1=0 THEN 540
920   E1=E1+1
930   IF R=1 THEN 540
940   GOTO 2650
950   !****** END OF MAIN PROGRAM ***********************************************
960   !
970   !****** SUBROUTINE READ AND POSITION S/V PORTS ***************************
980   !>>>>>>SCANIVALVE #1 HAS INOP HOME FEATURE<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
990 Read:   OUTPUT 707 USING "#,K";V
1000  P0=SPOLL(707)
1010  L=BINAND(P0,15)
1020  T1=ROTATE(P0,4)
1030  M1=BINAND(T1,7)
1040  P=10*M1+L
1050  CLEAR 707
1060  IF P=A THEN Finish
1070  OUTPUT 701;"C"
1080  OUTPUT 701 USING "DDD";V-1
1090  OUTPUT 701;"C"
1100  WAIT .1
1110  GOTO Read
1120 Finish:  RETURN
1130  !******SUBROUTINE TO BARO PRESS & TEMPERATURES ***************************
1140  OUTPUT 701;"C"                   !Clear scanner 1
1150  OUTPUT 722;"F1R1M0Z1T3"          !Sets-up HP3456A DVM
1160  !>>>>>> BARO PRESS IS MANUAL INPUT <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
1170  INPUT "ENTER BAROMETRIC PRESSURE IN INCHES HG",X(1)
1180  ! INPUT "ENTER RPM",Y(6)
1190  OUTPUT 705;"INAU1SM1"            !Sets-up HP5335A Counter
1200  ENTER 705;Hz                     !Collect data from counter
1210  Y(6)=Hz*60
1220  PRINT USING "/,K,6D.DD,10X,K,6D,/";"BAROMETER(IN. OF HG)=",X(1),"RPM=",Y(6)
1230  S=2                              !HP495A Scanner #2 designated
1240  PRINT USING "5X,K,DD,/,2X,K,8X,K";"SCANNER #",S,"CHAN","TEMP (R)"
1250  FOR C=1 TO 5                     !Collecting temperatures  (chls 1-5)
1260  OUTPUT 708;"C"
1270  OUTPUT 708 USING "DDD";C         !Scanner #2
1280  TRIGGER 722                      !Trigger DVM
1290  ENTER 722;V
```

Figure D3 (cont) TPL Program: TURBO1

```
1300  T(C)=V                          !Storing temperatures in T(*)
1310  NEXT C
1320  FOR C=25 TO 26                  !Collecting temp (chls 25-26)
1330  OUTPUT 708;"C"
1340  OUTPUT 708 USING "DDD";C        !Scanner #2
1350  TRIGGER 722                     !Trigger DVM
1360  ENTER 722;V
1370  T(C-7)=V                        !Storing temperatures in T(18),T(19)
1380  NEXT C
1390  OUTPUT 708;"C"
1400  CLEAR 722
1410  T(6)=T(3)
1420  T(7)=T(5)
1430  T(3)=T(2)+T(18)
1440  T(5)=T(4)+T(19)
1450  FOR C=1 TO 5
1460  Y(C)=FNT(T(C)*1000)+460         !Converts temp to deg Rankine
1470  PRINT USING "2X,DDD,8X,6D.D";C,Y(C)
1480  NEXT C
1510  RETURN
1520  !****** SUBROUTINE FOR DATA REDUCTION **********************************
1530  !   TURBINE STATIC OUTLET PRESSURE WAS CHANGED FROM
1540  !   ATMOSPHERIC TO A VALVE MEASURED IN THE TURBINE OUTLET FLOW
1550  !
1560  !    VARIABLES FOR THIS SECTION
1570  !        M IS STORAGE ARRAY
1580  !      Q=MATRIX OF PRESS(IN. OF HG)
1590  !     M(1)=TURBINE HEAD(DELTA P ORIFICE)
1600  !     M(2)=COMPRESSOR HEAD(DELTA P COMPRESSOR)
1610  !      Y1=ORIFICE EXPANSION FACTOR(TURBINE)
1620  !      Y2=ORIFICE EXPANSION FACTOR(COMPRESSOR)
1630  !      K1=DISCHARGE COEF FOR FEED ORIFICE (TURB)
1640  !      K2=DISCHARGE COEF FOR FEED ORIFICE (COMP)
1650  !      C9=C SUB P FOR AIR= 0.24
1660  !      C8=CONV FACTOR BTU->HP
1670  !      C7=(GAMMA-1/GAMMA) WITH GAMMA=1.4
1680  !      Z1=ORIFACE FACTOR (TURB)
1690  !      Z2=ORIFACE FACTOR (COMPRESSOR)
1700  !      A3=ALPHA FACTOR (THERMAL) IN FLOW EQN (TURB)
1710  !      A4=ALPHA FACTOR (THERMAL) IN FLOW EQN (COMP)
1720  !      K3=FACTOR IN FLOW EQN (TURB)
1730  !      K4=FACTOR IN FLOW EQN (COMP)
1740  !      F1=FIRST ITERATION OF FLOW RATE (TURBINE)
1750  !      F2=FIRST ITERATION OF FLOW RATE (COMPRESSOR)
1760  !     M(3)=TURB FLOW RATE
1770  !     M(4)=COMP FLOW RATE
1780  !     M(5)=TURB HP
1790  !     M(6)=COMP HP
1800  !     M(7)=TURB PRESS RATIO (IN/OUT)
1810  !     M(8)=COMP PRESS RATIO (IN/OUT)
1820  !     M(9)=TURB EFFICIENCY  (TOTAL-STATIC)
1830  !     M(10)=COMPRESSOR EFFICIENCY (TOTAL-TOTAL) MJ
1840  !      T0=REF TEMP (518.7 DEG R)
1850  !     M(11)=SQRT OF THETA (TURB)  (T(IN)/T(REF))^0.5
1860  !     M(12)=SQRT OF THETA (COMP)
1870  !      D0=REF PRESS (29.92 IN. HG)
1880  !     M(13)=REF PRESS RATIO (TURB IN/REF)
1890  !     M(14)=REP PRESS RATIO (COMP IN/REF)
1900  !     M(15)=REF TURB FLOW RATE
1910  !     M(16)=REF COMP FLOW RATE
1920  !     M(17)=REF TURB RPM
1930  !     M(18)=REF COMP RPM
1940  !     M(19)=REF TURB HP
1950  !     M(20)=REF COMP HP
1960  !     M(21)=DELTA TEMP(TURB)
```

Figure D3 (cont) TPL Program: TURBO1

```
1970  !     M(22)=DELTA TEMP(COMP)
1980  !     M(23)=HEAD COEFFICIENT
1990  MAT Q= (0)
2000  FOR I=6 TO 20 STEP 2
2010  Q(I/2-2)=((X(I)-X(2))*.07355)+X(1)
2020  NEXT I
2030  M(1)=X(6)-X(8)
2040  M(2)=X(14)-X(16)
2050  Y1=1-.026*M(1)/Q(1)
2060  Y2=1-.02581*M(2)/Q(5)
2070  K1=1.4405
2080  K2=.63094
2090  C9=.24
2100  C8=1.414836364
2110  C7=.2857142857
2120  Z1=1.9+C9*(((Y(1)-460)/100)-1)
2130  Z2=1.9+C9*(((Y(4)-460)/100)-1)
2140  A3=1+.002044*((Y(1)-528)/100)
2150  A4=1+.002044*((Y(4)-528)/100)
2160  K3=.00327673
2170  K4=.0015472
2180  F1=K1*A3*Y1*SQRT(ABS(Q(1)*M(1)/Y(1)))
2190  F2=K2*A4*Y2*SQRT(ABS(Q(5)*M(2)/Y(4)))
2200  M(3)=F1/2*(1+SQRT(ABS(1+4*K3*Z1/F1)))
2210  M(4)=F2/2*(1+SQRT(ABS(1+4*K4*Z2/F2)))
2220  M(21)=Y(2)-Y(3)
2230  M(22)=Y(5)-Y(4)
2240  M(5)=M(3)*C9*M(21)*C8
2250  M(6)=M(4)*C9*M(22)*C8
2260  M(7)=Q(3)/Q(4)
2270  M(8)=Q(8)/Q(7)
2280  M(9)=(M(21)/Y(2))*(1/(1-(Q(4)/Q(3))^C7))*100
2290  M(10)=Y(4)/M(22)*((M(8)^C7)-1)*100
2300  T0=518.7
2310  M(11)=SQRT(Y(2)/T0)
2320  M(12)=SQRT(Y(4)/T0)
2330  D0=29.92
2340  M(13)=Q(3)/D0
2350  M(14)=Q(7)/D0
2360  M(15)=M(3)*M(11)/M(13)
2370  M(16)=M(4)*M(12)/M(14)
2380  M(17)=Y(6)/M(11)
2390  M(18)=Y(6)/M(12)
2400  M(19)=M(5)/(M(13)*M(11))
2410  M(20)=M(6)/(M(14)*M(12))
2420  M(23)=12872868*Y(2)/Y(6)^2*(1-(Q(4)/Q(3))^C7)
2430  PRINT
2440  PRINT
2450  PRINT "REDUCED DATA:"
2460  PRINT
2470  PRINT "HEAD COEFFICIENT=";M(23)
2480  PRINT USING "/,8X,K";"HP       FLOW     DELTA     PRESS      EFF       FLOW     RPM     HP"
2490  PRINT USING "17X,K";"RATE     TEMP     RATIO              (REF)    (REF)  (REF)"
2500  !PRINT "KKKK_XXXXX.X_XXXXXX.XXX_XXXXXX.X_XXXXXX.XXX_XXXXXX.X_XXXXXX.XXX_XXXXXX_XXXXX.X"
2510  PRINT USING "/,K,X,5D.D,X,6D.3D,X,6D.D,X,6D.3D,X,6D.D,X,6D.3D,X,6D,X,5D.D";"TURB",M(5),M(3)
2520  PRINT USING "/,K,X,5D.D,X,6D.3D,X,6D.D,X,6D.3D,X,6D.D,X,6D.3D,X,6D,X,5D.D";"COMP",M(6),M(4)
2530  !
2540  RETURN
2550  !**** SUBROUTINE TO SAVE DATA POINT ON HARD DRIVE :,700,0,1 *************
2560 Save: OUTPUT @Path1;AS
2570  OUTPUT @Path1;E1
2580  OUTPUT @Path1;X(*),Y(*),T(*)
2590  OUTPUT @Path2;BS
2600  OUTPUT @Path2;E1
2610  OUTPUT @Path2;M(*)
```

Figure D3 (cont) TPL Program: TURBO1

```
2620  RETURN
2630  REM**** END OF SUBROUTINES ***********************************************
2640  !
2650  LOAD "TURBO_LAB",10
2660  END
2670  !
2680  REM**** FUNCTION FOR TEMPERATURE COMPUTATION ***************************
2690  DEF FNT(V)
2700  S1=32.144+35.77*V-.4518*V^2
2710  S2=33.252+34.86*V-.1855*V^2
2720  IF S1<100 THEN 2740
2730  S1=S2
2740  RETURN S1
2750  FNEND
2760  REM**** END OF USER DEFINED FUNCTIONS ***********************************
```

Figure D3 (cont) TPL Program: TURBO1

```
10    !  PROGRAM:  TURBO2  (OLD TURBO3)
20    !  MODIFIED BY:  R. SHREEVE 10/8/86
30    !                R. WENDLAND, LCDR, USN, 9/17/91
40    CLEAR SCREEN
50    DIM X(20),Y(6),T(19),IS[12],OS[12],US[14]
60    INTEGER E1
70    PRINTER IS CRT
80    INPUT "ENTER RAW DATA FILE NAME:",US
90    INPUT "ENTER DATA POINT RECORD NUMBERS (LOW, HIGH):",A,B
100   INPUT "DO YOU WANT TO DISPLAY RESULTS TO SCREEN OR PRINTER (SCREEN=0 PRINTER=1)?",R
110   IF R=0 THEN 130
120   PRINTER IS 711
130   ASSIGN @Path TO US&":,700,0,1"
140   ON END @Path GOTO Continue1
150   GOSUB Read
160   IS="  INLET"
170   OS="  OUTLET"
180   PRINT "-------------------------------------------------------------------------------"
190   PRINT TAB(23);"CUMMINS TURBOCHARGER MEASURED DATA"
200   PRINT "-------------------------------------------------------------------------------"
210   PRINT USING "16X,K,DD,19X,K,DD,K,DD,K,DD";"RUN #",X(9),"DATE: ",X(3),"/",X(5),"/",X(7)
220   PRINT "-------------------------------------------------------------------------------"
230   PRINT USING "/,/,27X,K";"PRESSURES (IN. H2O GAUGE)"
240   PRINT USING "20X,K,5X,K";"|   ORIFICE    |   TURBINE    |   NOZZLE","|  COMPRESSOR"
250   PRINT "POINT  TARE  CALIB. |";TAB(36);"|";TAB(51);"|";TAB(66);"|"
260   PRINT USING "K,14X,K,K,K,K,K,K,K,K";" NO. ",IS,OS,IS,OS,IS,OS,IS,OS
270   PRINT USING "10D,6D,8D,7D,8D,7D,8D,7D,8D,8D";1,2,3,4,5,6,7,8,9,10
280   PRINT "-------------------------------------------------------------------------------"
290 Print1:  IF E1<A THEN Loop1
300   PRINT USING "3D,6D.D,5D.D,6D.D,5D.D,6D.D,5D.D,6D.D,5D.D,6D.D,5D.D";X(11),X(2),X(4),X(6),X(8)
310   IF E1=B THEN Continue1
320 Loop1:GOSUB Read
330   GOTO Print1
340 Continue1:  PRINT "-------------------------------------------------------------------------------
350   ASSIGN @Path TO US&":,700,0,1"
360   ON END @Path GOTO Continue2
370   E1=0
380   GOSUB Read
390   IS="     INLET"
400   OS="     OUTLET"
410   PRINT USING "/,/,47X,K";"TEMPERATURES (DEG R)"
420   PRINT USING "28X,K";"|"
430   PRINT USING "K,5X,K,4X,K,10X,K,6X,K,7X,K,4X,K";"POINT","RPM","ATM. PRESS |","|","TURBINE","
440   PRINT USING "K,15X,K,K,K,K,K";" NO.";"(IN.HG)    ORIFICE",IS,OS,IS,OS
450   PRINT USING "23X,11D,11D,10D,11D,10D";1,2,3,4,5
460   PRINT "-------------------------------------------------------------------------------"
470 Print2:  IF E1<A THEN Loop2
480   PRINT USING "3D,11D,8D.DD,9D.D,9D.D,8D.D,9D.D,8D.D";X(11),Y(6),X(1),Y(1),Y(2),Y(3),Y(4),Y(5)
490   IF E1=B THEN Continue2
500 Loop2:GOSUB Read
510   GOTO Print2
520 Continue2:  PRINT "-------------------------------------------------------------------------------
530   PRINT TAB(21);"NOTE: CHANNEL AND PORT NUMBERS INDICATED"
540   PRINT "-------------------------------------------------------------------------------"
550   GOTO Hold
560 Read:  ENTER @Path;A$
570   ENTER @Path;E1
580   ENTER @Path;X(*)
590   ENTER @Path;Y(*)
600   ENTER @Path;T(*)
610   RETURN
620 Hold:DISP "F2 TO CONTINUE"
630   PAUSE
640   PRINTER IS CRT
650   LOAD "TURBO_LAB",10
660   END
```

Figure D4 TPL Program: TURBO2

```
10    !  PROGRAM:  TURBO3  (OLD TURBO4)
20    !  MODIFIED BY:  R. SHREEVE 10/8/86
30    !                R. WENDLAND, LCDR, USN, 9/20/91
40    CLEAR SCREEN
50    DIM X(20),Y(6),T(19),M(23),US[14],VS[14]
60    INTEGER E1,E2
70    PRINTER IS CRT
80    INPUT "ENTER RAW DATA FILE NAME:",US
90    INPUT "ENTER REDUCED DATA FILE NAME:",VS
100   INPUT "ENTER DATA POINT RECORD NUMBERS (LOW, HIGH):",A,B
110   INPUT "DO YOU WANT TO DISPLAY RESULTS TO SCREEN OR PRINTER (SCREEN=0 PRINTER=1)?",R
120   IF R=0 THEN 140
130   PRINTER IS 711
140   ASSIGN @Path1 TO US&":,700,0,1"
150   ASSIGN @Path2 TO VS&":,700,0,1"
160   ON END @Path1 GOTO Continue1
170   ON END @Path2 GOTO Continue1
180   GOSUB Read
190   PRINT "-------------------------------------------------------------------------------"
200   PRINT TAB(23);"CUMMINS TURBOCHARGER CALCULATED DATA"
210   PRINT "-------------------------------------------------------------------------------"
220   PRINT USING "16X,K,DD,19X,K,DD,K,DD,K,DD";"RUN #",X(9),"DATE: ",X(3),"/",X(5),"/",X(7)
230   PRINT "-------------------------------------------------------------------------------"
240   PRINT USING "/,/,32X,K";"TURBINE DATA"
250   PRINT USING "9X,K";"POINT   PRESS    REF      REF      HP      REF   HEAD     EFF"
260   PRINT USING "10X,K";"NO.    RATIO    RPM      FLOW             HP    COEFF    (%)"
270   PRINT USING "16X,K,9X,K";"(P6/P5)","(LBM/SEC)"
280   PRINT USING "10X,K";"       M(7)   M(17)     M(15)   M(5)    M(19)  M(23)   M(9)"
290   PRINT "-------------------------------------------------------------------------------"
300 Print1:  IF E1<A THEN Loop1
310   PRINT USING "9X,3D,6D.2D,9D,5D.3D,6D.2D,5D.2D,3D.3D,5D.D";X(11),M(7),M(17),M(15),M(5),M(19),
320   IF E1=B THEN Continue1
330 Loop1:GOSUB Read
340   GOTO Print1
350 Continue1:  PRINT "-------------------------------------------------------------------------------
360   ASSIGN @Path1 TO US&":,700,0,1"
370   ASSIGN @Path2 TO VS&":,700,0,1"
380   ON END @Path1 GOTO Continue2
390   ON END @Path2 GOTO Continue2
400   E1=0
410   GOSUB Read
420   PRINT USING "/,/,30X,K";"COMPRESSOR DATA"
430   PRINT USING "9X,K";"POINT   PRESS    RPM      REF      REF      HP      REF     EFF"
440   PRINT USING "10X,K";"NO.    RATIO            RPM      FLOW             HP      (%)"
450   PRINT USING "15X,K,16X,K";"(P10/P9)","(LBM/SEC)"
460   PRINT USING "10X,K";"       M(8)           M(18)   M(16)   M(6)   M(20)  M(10)"
470   PRINT "-------------------------------------------------------------------------------"
480 Print2:  IF E1<A THEN Loop2
490   PRINT USING "9X,3D,6D.2D,9D,8D,4D.3D,5D.2D,4D.2D,4D.D";X(11),M(8),Y(6),M(18),M(16),M(6),M(20)
500   IF E1=B THEN Continue2
510 Loop2:GOSUB Read
520   GOTO Print2
530 Continue2:  PRINT "-------------------------------------------------------------------------------
540   GOTO Hold
550 Read:  ENTER @Path1;A$
560   ENTER @Path1;E1
570   ENTER @Path1;X(*)
580   ENTER @Path1;Y(*)
590   ENTER @Path1;T(*)
600   ENTER @Path2;B$
610   ENTER @Path2;E2
620   ENTER @Path2;M(*)
630   RETURN
640 Hold:DISP "F2 TO CONTINUE"
650   PAUSE
```

Figure D5 TPL Program: TURBO3

```
660   PRINTER IS CRT
670   LOAD "TURBO_LAB",10
680   END
```

Figure D5 (cont) TPL Program: TURBO3

```
10    !Program:  TURBO4 Plots data from designated ASCII files to the CRT or Printer
20    !Developed:  by R. Wendland, LCDR, USN,  22 Sept 1991
30    !
40    DIM Title$[50],X_label$[50],Y_label$[50],Red_file$[14],M(23),Data1(1:20,23),Data2(1:20,23)
50    CLEAR SCREEN
60    !EY LABELS OFF
70    PRINTER IS CRT
80    !
90    !---- Load working matrix Data: ----------------------------------------
100 Load_matrix:  !
110   INPUT "Number of plots per graph (1 to 3):",Num_plots
120   FOR K=1 TO Num_plots
130    INPUT "Enter reduced data file name:",Red_file$
140    PRINT "Data file #";K;":  ";Red_file$
150    ASSIGN @Path TO Red_file$&":,700,0,1"
160    ON END @Path GOTO 410
170    I=1
180 Read:ENTER @Path;A$
190    ENTER @Path;E1
200    ENTER @Path;M(*)
210    SELECT K
220    CASE 1
230     FOR J=0 TO 23
240      Data1(I,J)=M(J)
250     NEXT J
260     D1=D1+1
270    CASE 2
280     FOR J=0 TO 23
290      Data2(I,J)=M(J)
300     NEXT J
310     D2=D2+1
320    CASE 3
330     FOR J=0 TO 23
340      Data3(I,J)=M(J)
350     NEXT J
360     D3=D3+1
370    END SELECT
380    I=I+1
390    GOTO Read
400    I=0
410   NEXT K
420   !
430   !---- Define graph parameters: -----------------------------------------
440 Graph_parameter:  !
450   INPUT "Enter graph title:",Title$
460   PRINT "Graph title:  ";Title$
470   INPUT "Enter X-axis label:",X_label$
480   PRINT "X-axis label:  ";X_label$
490   INPUT "Enter Y-axis label:",Y_label$
500   PRINT "Y-axis label:  ";Y_label$
510   INPUT "Enter X-axis and Y-axis M-values( M(x),M(y) ):",X,Y
520   FOR K=1 TO Num_plots
530    SELECT K
540    CASE 1
550     PRINT
560     PRINT "Data file #";K;":"
570     PRINT X_label$;TAB(25);Y_label$
580     FOR I=1 TO D1
590      PRINT Data1(I,X),Data1(I,Y)
600     NEXT I
610    CASE 2
620     PRINT
630     PRINT "Data file #";K;":"
640     PRINT X_label$;TAB(25);Y_label$
650     FOR I=1 TO D2
```

Figure D6 TPL Program: TURBO4

```
660       PRINT Data2(I,X),Data2(I,Y)
670      NEXT I
680     CASE 3
690      PRINT
700      PRINT "Data file #";K;":"
710      PRINT X_label$;TAB(25);Y_label$
720      FOR I=1 TO D3
730       PRINT Data3(I,X),Data3(I,Y)
740      NEXT I
750     END SELECT
760    NEXT K
770    INPUT "Enter Xmin, Xmax, Ymin, Ymax:",Xo,Xf,Yo,Yf
780    !
790    !---- Graph data ----------------------------------------------------
800 Graph_data:  !
810    CLEAR SCREEN
820    GINIT                          !Initialize graph routine
830    Xso=Xo                         ! Initialize all scaling variables
840    Xsf=Xf
850    Yso=Yo
860    Ysf=Yf
870    Xs_range=Xsf-Xso
880    Ys_range=Ysf-Yso
890    Xscale=1
900    Yscale=1
910    IF Xs_range<1000 THEN          ! Scale up graph variables for mininum of
920     Xs_range=Xs_range*10          ! of 1000 units of resolution across x-axis
930     Xscale=Xscale*10
940     GOTO 910
950    END IF
960    IF Ys_range<1000 THEN          ! Scale up graph variables for mininum of
970     Ys_range=Ys_range*10          ! of 1000 units of resolution across y-axis
980     Yscale=Yscale*10
990     GOTO 960
1000   END IF
1010   Xso=Xo*Xscale                  ! Finalize all scaling variables
1020   Xsf=Xf*Xscale
1030   Yso=Yo*Yscale
1040   Ysf=Yf*Yscale
1050   Xs_range=Xsf-Xso                 !Length of X-axis
1060   Ys_range=Ysf-Yso                 !Length of Y-axis
1070   LORG 6                           !Character ref pt:top center
1080   MOVE 100*RATIO/2,100             !Move cursor to screen loc for labels
1090   LABEL Title$                     !Plot title
1100   CSIZE 3.5                        !Sizes labeling
1110   MOVE 100*RATIO/2,0               !Move cursor to bottom center screen
1120   LORG 4                           !Character ref pt:bottom center
1130   LABEL X_label$                   !X-axis label
1140   DEG                              !Desig degrees for LDIR
1150   LDIR 90                          !Sets Y-axis label on end
1160   LORG 6
1170   MOVE 0,50
1180   LABEL Y_label$                   !Y-axis label
1190   LDIR 0                           !Reset label to horizontal orientation.
1200   LORG 2                           !Chr ref pt:left center
1210   VIEWPORT 10,90*RATIO,10,90       !Sets graph screen size
1220   FRAME                            !Box around VIEWPORT
1230   WINDOW Xso,Xsf,Yso,Ysf           !Set axis lengths in VIEWPORT
1240   AXES Xs_range/10,Ys_range/10,Xso,Yso     !Axes intersect at lower left
1250   AXES Xs_range/10,Ys_range/10,Xsf,Ysf     !Axes intersect at upper right
1260   CLIP OFF                         !So labels can print outside VIEWPORT
1270   CSIZE 3.0,.4                     !Axes label size
1280   LORG 8                           !Number X-axis
1290   FOR I=Xso TO Xsf STEP Xs_range/10
1300    MOVE I,Yso-.01*Ys_range
```

Figure D6 (cont) TPL Program: TURBO4

```
1310   LABEL USING "#,K";I/Xscale
1320  NEXT I
1330  LORG 8                              !Number Y-axis
1340  FOR I=Yso TO Ysf STEP Ys_range/10
1350   MOVE Xso-.01*Xs_range,I
1360   LABEL USING "#,K";I/Yscale
1370  NEXT I
1380  FOR K=1 TO Num_plots
1390   SELECT K
1400   CASE 1
1410    FOR I=1 TO D1
1420     PLOT Data1(I,X)*Xscale,Data1(I,Y)*Yscale,-2  ! Plots Data1 as squares
1430     PDIR 45
1440     POLYGON .01*Xs_range,4
1450    NEXT I
1460   CASE 2
1470    FOR I=1 TO D2
1480     PLOT Data2(I,X)*Xscale,Data2(I,Y)*Yscale,-2  ! Plots Data2 as diamonds
1490     PDIR 0
1500     POLYGON .01*Xs_range,4
1510    NEXT I
1520   CASE 3
1530    FOR I=1 TO D3
1540     PLOT Data3(I,X)*Xscale,Data3(I,Y)*Yscale,-2  ! Plots Data3 as triangles
1550     PDIR 90
1560     POLYGON .01*Xs_range,3
1570    NEXT I
1580   END SELECT
1590  NEXT K
1600  IF Plot=0 THEN Hard_copy
1610  DUMP GRAPHICS #711
1620  GOTO Another_plot
1630  !
1640  !---- Hard copy plots of graphs ------------------------------------------
1650 Hard_copy: !
1660  INPUT "Plot graph: (No=0 Printer=1 Plotter=2)",Plot
1670  IF Plot=0 THEN Another_plot
1680  IF Plot=1 THEN Graph_data
1690  !---- Plotter rout'       ------------------------------------------------
1700 Plotter: !
1710  PRINTER IS 712
1720  Xso=Xo                          ! Initialize all scaling variables
1730  Xsf=Xf
1740  Yso=Yo
1750  Ysf=Yf
1760  Xs_range=Xsf-Xso
1770  Ys_range=Ysf-Yso
1780  Xscale=1
1790  Yscale=1
1800  IF Xs_range<1000 THEN           ! Scale up graph variables for mininum of
1810   Xs_range=Xs_range*10           ! of 1000 units of resolution across x-axis
1820   Xscale=Xscale*10
1830   GOTO 1800
1840  END IF
1850  IF Ys_range<1000 THEN           ! Scale up graph variables for mininum of
1860   Ys_range=Ys_range*10           ! of 1000 units of resolution across y-axis
1870   Yscale=Yscale*10
1880   GOTO 1850
1890  END IF
1900  Xsf=INT(Xf*Xscale)
1910  IF Xsf>16383 THEN               ! Readjust graph variables to ensure x-axis
1920   Xsf=Xsf/10                     ! Xsf does not exceed 16383 units
1930   Xscale=Xscale/10
1940   GOTO 1910
1950  END IF
```

Figure D6 TPL Program: TURBO4

```
1960  Ysf=INT(Yf*Yscale)
1970  IF Ysf>16383 THEN              ! Readjust graph variables to ensure y-axis
1980   Ysf=Ysf/10                    ! Xsf does not exceed 16383 units
1990   Yscale=Yscale/10
2000   GOTO 1970
2010  END IF
2020  Xso=INT(Xo*Xscale)             ! Finalize all scaling variables
2030  Xsf=INT(Xf*Xscale)
2040  Yso=INT(Yo*Yscale)
2050  Ysf=INT(Yf*Yscale)
2060  PRINT "IN;IP1000,1000,9500,7000;"          ! Set paper size
2070  PRINT "SC";Xso;",";Xsf;",";Yso;",";Ysf;";"   ! Set X & Y axises lengths
2080  PRINT "PA";Xso;",";Yso;";SP1;PD;"
2090  FOR Xp=Xso TO Xsf STEP (Xsf-Xso)/10          ! Print X-axis
2100   PRINT "PA";Xp;",";Yso;";XT;PU;"
2110   PRINT "CP-2.4,-1.3;LB";Xp/Xscale;CHR$(3)
2120   PRINT "PA";Xp;",";Yso;";PD;"
2130  NEXT Xp
2140  PRINT "PU;PA";Xso+INT(.4*(Xsf-Xso));",";Yso-INT(.078*(Ysf-Yso));";LB";X_label$;CHR$(3)
2150  PRINT "PU;PA";Xso;",";Ysf;";PD;"
2160  FOR Xp=Xso TO Xsf STEP (Xsf-Xso)/10
2170   PRINT "PA";Xp;",";Ysf;";XT;"
2180  NEXT Xp
2190  PRINT "PU;PA";Xso;",";Yso;";PD;"
2200  FOR Yp=Yso TO Ysf STEP (Ysf-Yso)/10          ! Print Y-axis
2210   PRINT "PA";Xso;",";Yp;";YT;PU;"
2220   PRINT "CP-6.4,-.3;LB";Yp/Yscale;CHR$(3)
2230   PRINT "PA";Xso;",";Yp;";PD;"
2240  NEXT Yp
2250  PRINT "PU;PA";Xso-INT(.09*(Xsf-Xso));",";Yso+INT(.375*(Ysf-Yso));";DI0,1;LB";Y_label$;CHR$(3)
2260  PRINT "PA";Xsf;",";Yso;";PD"
2270  FOR Yp=Yso TO Ysf STEP (Ysf-Yso)/10
2280   PRINT "PA";Xsf;",";Yp;";YT;"
2290  NEXT Yp
2300  PRINT "PU;PA";Xso+INT(.4*(Xsf-Xso));",";INT(.078*(Ysf-Yso))+Ysf;";DI1,0;LB";Title$;CHR$(3)
2310  FOR K=1 TO Num_plots                         ! Plot data points
2320   SELECT K
2330   CASE 1
2340    FOR I=1 TO D1
2350     PRINT "PA";INT(Data1(I,X)*Xscale);",";INT(Data1(I,Y)*Yscale);";"
2360     PRINT "UC-3,-4,99,0,8,6,0,0,-6,-6,0,-99;" ! Plots squares for graph #1
2370    NEXT I
2380   CASE 2
2390    FOR I=1 TO D2
2400     PRINT "PA";INT(Data2(I,X)*Xscale);",";INT(Data2(I,Y)*Yscale);";"
2410     PRINT "UC0,-4,99,-3,4,3,4,3,-4,-3,-4,-99;"! Plots diamonds for graph #2
2420    NEXT I
2430   CASE 3
2440    FOR I=1 TO D3
2450     PRINT "PA";INT(Data3(I,X)*Xscale);",";INT(Data3(I,Y)*Yscale);";"
2460     PRINT "UC-3,-4,99,3,8,3,-8,-6,0,-99;"    ! Plots triangles for graph #3
2470    NEXT I
2480   END SELECT
2490  NEXT K
2500  PRINT "PU;PA";Xsf;",";Ysf;";"
2510  PRINTER IS CRT
2520  !---- Additionl plots/End of routines ----------------------------------
2530 Another_plot: !
2540  INPUT "Plot another graph? (Yes=1 No=0)",Replot
2550  Plot=0
2560  CLEAR SCREEN
2570  IF Replot=1 THEN Graph_parameter
2580  CLEAR SCREEN
2590  KEY LABELS ON
2600  LOAD "TURBO_LAB",10
```

Figure D6 TPL Program: TURBO4

```
10    !Program: SCAN_TEMP
20    !Description:  Reads thermo-couple emf from channels 0-5, converts to
30    !              degrees F, and prints to CRT.
40    !Devices:  HP3495A(708), HP3456A(722)
50    !
60    CLEAR SCREEN
70    DUMP DEVICE IS 711
80    DIM Y(5,1:2),T(5)
90    INTEGER Run
100   DIM Channel$(5)[20]
110   Channel$(0)="Atmosphere"
120   Channel$(1)="Supply Orifice"
130   Channel$(2)="Turbine Inlet"
140   Channel$(3)="Turbine Outlet"
150   Channel$(4)="Compressor Inlet"
160   Channel$(5)="Compressor Outlet"
170   !
180   PRINT "Turbocharger Lab Temperature Readings."
190 Scan:  !-----------------------------------------------------------------
200   Run=Run+1
210   PRINT
220   PRINT "Run number:";Run
230   PRINT USING "K,11X,K,3X,K";"Location","Channel #","   Temperature (F)"
240   PRINT
250   OUTPUT 722;"T3"
260   FOR C=0 TO 5                    !Reading temperature channels 0-5
270     OUTPUT 708 USING "DDD";C
280     TRIGGER 722
290     ENTER 722;T(C)
300     Y(C,1)=C
310     Y(C,2)=FNT(T(C)*1000)         !User defined function applied
320     IF Y(C,2)<ABS(.1) THEN
330       Y(C,2)=0
340     END IF
350     PRINT USING "20A,2X,DD,11X,5D.D";Channel$(C),Y(C,1),Y(C,2)
360   NEXT C
370   CLEAR 722
380   INPUT "Enter: 1=Rescan  0=Quit",Rescan
390   IF Rescan=1 THEN Scan
400   LOAD "TURBO_MENU",10
410   END!=============================================================
420   !User defined function to convert voltage readings from temperature probes
430   !to temperature values in degrees F.
440   DEF FNT(V)
450     S1=32.144+35.77*V-.4518*V^2
460     S2=33.252+34.86*V-.1855*V^2
470     IF S1<100 THEN 490
480     S1=S2
490     RETURN S1
500   FNEND
```

Figure D7 TPL Program: SCAN_TEMP

```
10    !Program: TURBO_MENU
20    !Description:  Program provides Function Key menu for program selection.
30    CLEAR SCREEN
40    ON KEY 1 LABEL "TURB01" GOTO Turbo1
50    ON KEY 2 LABEL "TURB02" GOTO Turbo2
60    ON KEY 3 LABEL "TURB03" GOTO Turbo3
70    ON KEY 4 LABEL "TURB04" GOTO Turbo4
80    ON KEY 5 LABEL "SCAN    TEMP " GOTO Temp
90    ON KEY 6 LABEL " " GOTO Hold
100   ON KEY 7 LABEL "MAIN    MENU" GOTO Main
110   ON KEY 8 LABEL "EXIT    MENU" GOTO Exit
120   !
130   PRINT "Turbocharger Data Acquisition Lab"
140   PRINT
150   PRINT "Item:                              Select Function Key"
160   PRINT
170   PRINT "  TURBO1:  Collect, reduce and              F1"
180   PRINT "           store RAW Data"
190   PRINT
200   PRINT "  TURBO2:  Tabulate RAW Data                F2"
210   PRINT
220   PRINT "  TURBO3:  Tabulate REDUCED Data            F3"
230   PRINT
240   PRINT "  TURBO4:  Plot REDUCED Data                F4"
250   PRINT
260   PRINT "  SCAN_TEMP: Scan temperature probs         F5"
270   PRINT
280   PRINT "  Main Menu                                 F7"
290   PRINT "  Exit Menu                                 F8"
300   !
310 Hold:    !
320   GOTO Hold
330 Turbo1:  !
340   LOAD "TURBO1",10
350 Turbo2:  !
360   LOAD "TURBO2",10
370 Turbo3:  !
380   LOAD "TURBO3",10
390 Turbo4:  !
400   LOAD "TURBO4",10
410 Temp:  !
420   LOAD "SCAN_TEMP",10
430 Discrepancies:!
440   LOAD "DISCREP_DOC",10
450 Main:    !
460   MASS STORAGE IS "/WORKSTATIONS"
470   LOAD "AUTOST",10
480 Exit:CLEAR SCREEN
490   END
```

Figure D8 TPL Program: TURBO_MENU

```
10    !Title:  A_4431T
20    !Author:  R.P. Shreeve (Rev. 12/3/89)
30    !Updated:  R.A. Wendland (12/12/91)
40    !Program:  Performance calculations for turbine for varying
50    !          Alpha1.
60    !
70    CLEAR SCREEN
80    DUMP DEVICE IS 711
90    KEY LABELS OFF
100   DIM A(7,15),B(7,15),C(7,15),R(15),S(15),Alpha1(15)
110   DEG                                ! All angles in degrees
120   INPUT "Print results to CRT or Printer? (CRT=1, Printer=0): ",Crt
130   IF Crt=1 THEN 150
140   PRINTER IS 711
150   !----- Input data ---------------------------------------------------
160   R9=.08
170   P9=1
180   Q9=517.8
190   P0=2
200   P2=1
210   T0=560
220   U0=800
230   H1=14
240   ! A1=75                           ! Program modified for Alpha1 iteration
250   G1=1.4
260   G0=32.174
270   K9=.98
280   N0=35000
290   INPUT "Enter R*: ",R0
300   INPUT "Enter Alpha1 (deg): min, max, step size ==> ",L1,L2,L3
310   !----- Calculate needed quantities ----------------------------------
320   V0=109.62*SQRT(T0)
330   X3=U0/V0
340   P4=P2/P0
350   R8=R9*(P0/P9)/(T0/Q9)
360   !----- Calculate performance ----------------------------------------
370   I=1
380   L4=0
390   FOR A1=L1 TO L2 STEP L3
400   GOSUB Non_dimension
410   T8=T9*T0
420   H0=(V0*V0/(2*G0))*T9/550
430   P1=P3*P0
440   T1=S1*T0
450   T2=S2*T0
460   V1=X1*V0
470   V2=X2*V0
480   W1=X5*V0
490   W2=X6*V0
500   W0=H1/H0
510   X9=X1*COS(A1)
520   A9=(W0*144/(R8*V0))/(X9*(1-X9^2)^(1/(1-G1))*K9)
530   R7=(360*U0)/(PI*N0)
540   H9=A9/(2*PI*R7)
550   PRINT "R*=";R0;"RM=";R7;"H=";H9;"W0=";W0
560   !----- Store data in array ------------------------------------------
570   R(I)=R0
580   S(I)=R1
590   Alpha1(I)=A1
600   A(1,I)=N1
610   A(2,I)=N2
620   A(3,I)=H0
630   A(4,I)=T8
640   A(5,I)=P1
650   A(6,I)=T1
```

Figure D9 TPL Program: A_4431T

```
660   A(7,I)=T2
670   B(1,I)=V1
680   B(2,I)=W1
690   B(3,I)=V2
700   B(4,I)=W2
710   B(5,I)=B1
720   B(6,I)=-A2
730   B(7,I)=-B2
740   C(1,I)=Z1
750   C(2,I)=Z2
760   C(3,I)=M3
770   C(4,I)=M4
780   C(5,I)=M5
790   C(6,I)=M6
800   C(7,I)=0
810   I=I+1
820   L4=L4+1
830   NEXT A1
840   !----- Print data ----------------------------------------------------
850   PRINT USING "/,K,5D.2D,K,5D.2D,K,6D,K";"PT0=",P0,"      P2=",P2," ATO.      TT0=",T0," DEG. R"
860   PRINT USING "/,K,6D,K,/";"WHEEL VEL.=",U0," FT/SEC."
870   FOR I=1 TO L4
880   PRINT USING "K,5D";"ALPHA1=",Alpha1(I)
890   NEXT I
900   PRINT USING "/,K,K,K,K,K,K,K,K,K";" ETA T-S"," ETA T-T","  HP/LB ","  DELTW ","     P1 ","
910   FOR I=1 TO L4
920   PRINT USING "4D.3D,4D.3D,3D.4D,3D.4D,3D.4D,5D.2D,5D.2D,5D.2D,3D.4D";A(1,I),A(2,I),A(3,I),A(
930   NEXT I
940   PRINT USING "/,K,K,K,K,K,K,K,K,K";"    V1  ","    W1  ","    V2  ","    W2  ","   BETA1 ","
950   FOR I=1 TO L4
960   PRINT USING "6D.D,6D.D,6D.D,6D.D,6D.D,6D.D,6D.D,5D.2D,3D.4D";B(1,I),B(2,I),B(3,I),B(4,I),B(
970   NEXT I
980   PRINT USING "/,K,K,K,K,K,K,K,K,K";"  ZETA S","  ZETA R","  M1CRIT","  M2CRIT","  MR1CRIT","
990   FOR I=1 TO L4
1000  PRINT USING "3D.4D,3D.4D,3D.4D,3D.4D,3D.4D,3D.4D,8X,5D.2D,3D.4D";C(1,I),C(2,I),C(3,I),C(4,I
1010  NEXT I
1020  GOTO 1410
1030  !----- Subroutine for non-dimension performance calculations -----
1040 Non_dimension:  !
1050  G2=(G1-1)/G1
1060  G3=SQRT((G1+1)/(G1-1))
1070  T4=P4^G2
1080  T3=T4+R0*(1-T4)
1090  P3=T3^(1/G2)
1100  Y1=1-T3
1110  M1=(2/(G1-1))*(Y1/(1-Y1))
1120  Z1=.1*M1
1130  S1=T3+Z1*Y1
1140  X1=SQRT(1-S1)
1150  X4=X1*COS(A1)
1160  B1=ATN((X1*SIN(A1)-X3)/X4)
1170  X5=X4/COS(B1)
1180  T7=S1+X5^2
1190  T5=S1*(P4/P3)^G2
1200  Y6=T7-T5
1210  M2=(2/(G1-1))*Y6/T5
1220  Z2=.2*M2
1230  S2=T5+Z2*Y6
1240  X6=SQRT(T7-S2)
1250  B2=X4/X6
1260  B2=ATN(SQRT(1-B2^2)/B2)
1270  A2=ATN((X6*SIN(B2)-X3)/X4)
1280  X2=X4/COS(A2)
1290  S3=S2+X2^2
1300  T9=1-S3
```

Figure D9 (cont) TPL Program: A_4431T

```
1310  N1=T9/(1-T4)
1320  P5=P4*(S3/S2)^(1/G2)
1330  T6=P5^G2
1340  N2=T9/(1-T6)
1350  R1=(S1-S2)/(1-S3)
1360  M3=X1*G3
1370  M4=X2*G3*SQRT(1/S3)
1380  M5=X5*G3*SQRT(1/T7)
1390  M6=X6*G3*SQRT(1/T7)
1400  RETURN
1410  DISP "Press F2 to exit, press F3 for another run"
1420  PRINTER IS CRT
1430  PAUSE
1440  LOAD "DESIGN_MENU",10
1450  END
```

Figure D9 (cont) TPL Program: A_4431T

```
10    !Title:  R_4431T
20    !Author:  R.P. Shreeve (Rev. 12/3/89)
30    !Updated:  R.A. Wendland (11/30/91)
40    !Program:  Performance calculations for turbine for varying
50    !          theoretical degree of reaction.
60    !
70    CLEAR SCREEN
80    DUMP DEVICE IS 711
90    KEY LABELS OFF
100   DIM A(7,15),B(7,15),C(7,15),R(15),S(15)
110   INPUT "Print results to CRT or Printer? (CRT=1, Printer=0): ",Crt
120   IF Crt=1 THEN 140
130   PRINTER IS 711
140   !----- Input data ---------------------------------------------------
150   R9=.08
160   P9=1
170   Q9=517.8
180   P0=2
190   P2=1
200   T0=560
210   U0=800
220   H1=14
230   A1=75
240   G1=1.4
250   G0=32.174
260   K9=.98
270   N0=35000
280   !----- Calculate needed quantities ----------------------------------
290   V0=109.62*SQRT(T0)
300   X3=U0/V0
310   P4=P2/P0
320   R8=R9*(P0/P9)/(T0/Q9)
330   DEG
340   !----- Calculate performance ----------------------------------------
350   INPUT "Enter R*: min, max, step size",L1,L2,L3
360   I=1
370   L4=0
380   FOR R0=L1 TO L2 STEP L3
390   GOSUB Non_dimension
400   T8=T9*T0
410   H0=(V0*V0/(2*G0))*T9/550
420   P1=P3*P0
430   T1=S1*T0
440   T2=S2*T0
450   V1=X1*V0
460   V2=X2*V0
470   W1=X5*V0
480   W2=X6*V0
490   W0=H1/H0
500   X9=X1*COS(A1)
510   A9=(W0*144/(R8*V0))/(X9*(1-X9^2)^(1/(1-G1))*K9)
520   R7=(360*U0)/(PI*N0)
530   H9=A9/(2*PI*R7)
540   PRINT "R*=";R0;"RM=";R7;"H=";H9;"W0=";W0
550   !----- Store data in array ------------------------------------------
560   R(I)=R0
570   S(I)=R1
580   A(1,I)=N1
590   A(2,I)=N2
600   A(3,I)=H0
610   A(4,I)=T8
620   A(5,I)=P1
630   A(6,I)=T1
640   A(7,I)=T2
650   B(1,I)=V1
```

Figure D10 TPL Program: R_4431T

```
660   B(2,I)=W1
670   B(3,I)=V2
680   B(4,I)=W2
690   B(5,I)=B1
700   B(6,I)=-A2
710   B(7,I)=-B2
720   C(1,I)=Z1
730   C(2,I)=Z2
740   C(3,I)=M3
750   C(4,I)=M4
760   C(5,I)=M5
770   C(6,I)=M6
780   C(7,I)=0
790   I=I+1
800   L4=L4+1
810   NEXT R0
820   !----- Print data ----------------------------------------------------
830   PRINT USING "/,/,/,/,K,5D.2D,K,5D.2D,K,6D,K";"PT0=",P0,"      P2=",P2," ATO.      TT0=",T0,"
840   PRINT USING "/,K,6D,K,5D,K";"WHEEL VEL.=",U0," FT/SEC.      ALPHA1=",A1," DEG."
850   PRINT USING "/,/,K,K,K,K,K,K,K,K,K";" ETA T-S"," ETA T-T","  HP/LB ","  DELTW ","     P1 ","
860   FOR I=1 TO L4
870   PRINT USING "4D.3D,4D.3D,3D.4D,3D.4D,3D.4D,5D.2D,5D.2D,5D.2D,3D.4D";A(1,I),A(2,I),A(3,I),A(4
880   NEXT I
890   PRINT USING "/,/,K,K,K,K,K,K,K,K,K";"    V1  ","    W1  ","    V2  ","    W2  ","   BETA1 ",
900   FOR I=1 TO L4
910   PRINT USING "6D.D,6D.D,6D.D,6D.D,6D.D,6D.D,6D.D,5D.2D,3D.4D";B(1,I),B(2,I),B(3,I),B(4,I),B(5
920   NEXT I
930   PRINT USING "/,/,K,K,K,K,K,K,K,K,K";"  ZETA S","  ZETA R","  M1CRIT","  M2CRIT"," MR1CRIT","
940   FOR I=1 TO L4
950   PRINT USING "3D.4D,3D.4D,3D.4D,3D.4D,3D.4D,3D.4D,8X,5D.2D,3D.4D";C(1,I),C(2,I),C(3,I),C(4,I)
960   NEXT I
970   GOTO 1360
980   !----- Subroutine for non-dimension performance calculations -----
990 Non_dimension:   !
1000  G2=(G1-1)/G1
1010  G3=SQRT((G1+1)/(G1-1))
1020  T4=P4^G2
1030  T3=T4+R0*(1-T4)
1040  P3=T3^(1/G2)
1050  Y1=1-T3
1060  M1=(2/(G1-1))*(Y1/(1-Y1))
1070  Z1=.1*M1
1080  S1=T3+Z1*Y1
1090  X1=SQRT(1-S1)
1100  X4=X1*COS(A1)
1110  B1=ATN((X1*SIN(A1)-X3)/X4)
1120  X5=X4/COS(B1)
1130  T7=S1+X5^2
1140  T5=S1*(P4/P3)^G2
1150  Y6=T7-T5
1160  M2=(2/(G1-1))*Y6/T5
1170  Z2=.2*M2
1180  S2=T5+Z2*Y6
1190  X6=SQRT(T7-S2)
1200  B2=X4/X6
1210  B2=ATN(SQRT(1-B2^2)/B2)
1220  A2=ATN((X6*SIN(B2)-X3)/X4)
1230  X2=X4/COS(A2)
1240  S3=S2+X2^2
1250  T9=1-S3
1260  N1=T9/(1-T4)
1270  P5=P4*(S3/S2)^(1/G2)
1280  T6=P5^G2
1290  N2=T9/(1-T6)
1300  R1=(S1-S2)/(1-S3)
```

Figure D10 (cont) TPL Program: R_4431T

```
1310  M3=X1*G3
1320  M4=X2*G3*SQRT(1/S3)
1330  M5=X5*G3*SQRT(1/T7)
1340  M6=X6*G3*SQRT(1/T7)
1350  RETURN
1360  DISP "Press F2 to exit, press F3 for another run"
1370  PAUSE
1380  LOAD "DESIGN_MENU",10
1390  END
```

Figure D10 (cont) TPL Program: R_4431T

```
10    !Title:  TURB3
20    !Author:  R. P. Shreeve (09/01/88)
30    !Updated:  R. Wendland, LCDR, USN (12/12/91)
40    !Program:  Calculate and draw turbine blade profiles.
50    !
60    DEG
70    DUMP DEVICE IS 711
80    CLEAR SCREEN
90    KEY LABELS OFF
100   !----- Input Graph parameters ----------------------------------------
110   INPUT "Plotting STATOR or ROTOR? (stator=0, rotor=1): ",Type
120   IF Type=1 THEN
130   Type$="ROTOR"
140   ELSE
150   Type$="STATOR"
160   END IF
170   PRINT "Input axis coordinates for ";Type$;" blade profile plot:"
180   IF Type=0 THEN
190   INPUT "Enter Xa (nominal -1.0): ",Xa
200   INPUT "Enter Xb (nominal 3.0): ",Xb
210   Type=1
220   ELSE
230   INPUT "Enter Xa (nominal 3.0): ",Xa
240   INPUT "Enter Xb (nominal -1.0): ",Xb
250   Type=-1
260   END IF
270   Ya=-.2
280   Yb=ABS(Xa-Xb)/RATIO+Ya
290   PRINT
300   PRINT "Axis coordinates:"
310   PRINT "  Xa = ",Xa
320   PRINT "  Xb = ",Xb
330   PRINT "  Ya = ",Ya
340   PRINT "  Yb = ",Yb
350   PRINT
360   !----- Input Data ----------------------------------------------------
370   PRINT "Enter all angles in degrees."
380   INPUT "Enter T.E. wedge angle (epsilon): ",E1
390   INPUT "Enter T.E. radius /spacing: ",R8
400   INPUT "Enter output relative flow angle (alpha2 or beta2): ",B3
410   B3=B3*Type
420   !----- EQ. 26 --------------------------------------------------------
430   A5=COS(B3)-8*R8*(1-B3/90)
440   A0=COS(B3)-8*R8*(1-ATN(SQR(1-A5^2)/A5)/90)
450   IF ABS(A0-A5)<.00001 THEN 480
460   A5=A0
470   GOTO 440
480   INPUT "Enter inlet rel. flow angle(alpha1 or beta1): ",B2
490   B2=-B2*Type
500   INPUT "Enter L.E. radius /spacing (le/S): ",R9
510   INPUT "Enter L.E. wedge angle (delta): ",D1
520   INPUT "Enter axial chord/spacing: ",B0
530   Z0=2*(TAN(B3)+TAN(B2))*COS(B3)^2/B0
540   S0=1
550   B1=B2-10
560   IF D1>20 THEN 580
570   B1=B2-D1/2
580   G1=B1+D1
590   A9=(A0+2*R8)/S0
600   E2=E1/2
610   A8=A9/(1+TAN(E2)^2)
620   A8=A8-SQR(A8^2-(A9^2-TAN(E2)^2)/(1+TAN(E2)^2))
630   A1=ATN(A8/SQR(1-A8^2))
640   R1=(B0-R8*(1+COS(A1+E1))-R9*(1+SIN(B1)))/(COS(A1+E1)+SIN(B1))
650   R2=S0*COS(A1+E1)/SIN(E1)+R8
```

Figure D11 TPL Program: TURB3

```
660   E0=1/(1+SIN(G1-A1-E1))
670   F0=(SIN(G1-A1-E1)+COS(D1))/(1+SIN(G1-A1-E1))
680   G0=(2*(COS(D1/2))^2)/(1+SIN(G1-A1-E1))
690   R3=A0*E0*(COS(G1)*(SIN(A1+E1)-S0/A0)-SIN(G1)*COS(A1+E1))+R1*F0+R9*G0
700   !----- PRINT DATA ------------------------------------------------
710   PRINTER IS 711
720   PRINT
730   PRINT "          Turbine Blade Profile"
740   PRINT "          ---------------------"
750   PRINT
760   PRINT "For  Axial Chord     = ";B0
770   PRINT "     Blade Spacing   = ";S0
780   PRINT "     Exit Width      = ";A0
790   PRINT "     L.E. Radius     = ";R9
800   PRINT "     T.E. Radius     = ";R8
810   PRINT "     L.E. Wedge Angle= ";D1
820   PRINT "     T.E. Wedge Angle= ";E1
830   PRINT
840   PRINT "Design Rel. Inlet Flow Angle  = ";-B2*Type
850   PRINT "Design Rel. Outlet Flow Angle = ";B3*Type
860   PRINT "Zweifel Coefficient           = ";Z0
870   PRINT
880   PRINT
890   PRINT "Calculated Blade Parameters:"
900   PRINT "----------------------------"
910   PRINT "     Alpha     = ";A1
920   PRINT "     Beta      = ";B1
930   PRINT "     Gamma     = ";G1
940   PRINT
950   PRINT "     R1        = ";R1
960   PRINT "     R2        = ";R2
970   PRINT "     R3        = ";R3
980   PRINT
990   PRINT
1000  PRINTER IS CRT
1010  DISP "Press F2 to view plot, F3 for new inputs and recalculate values."
1020  PAUSE
1030  CLEAR SCREEN
1040  !----- Calculate and Plot data to CRT screen --------------------------
1050  ! Graph parameters:
1060  X_range=Xb-Xa
1070  Y_range=Yb-Ya
1080  Dx=10
1090  Dy=10
1100  GINIT
1110  PEN 4
1120  LORG 6
1130  MOVE 100*RATIO/2,100
1140  LABEL Type$;" Blade Profiles"
1150  CSIZE 3.5
1160  MOVE 100*RATIO/2,0
1170  LORG 4
1180  LABEL "Meridian Plane"
1190  LDIR 90
1200  LORG 6
1210  MOVE 0,50
1220  LABEL "Chord"
1230  LDIR 0
1240  LORG 2
1250  VIEWPORT 10,95*RATIO,10,95
1260  FRAME
1270  WINDOW Xa,Xb,Ya,Yb
1280  AXES X_range/Dx,Y_range/Dy,Xa,Ya
1290  AXES X_range/Dx,Y_range/Dy,Xb,Yb
1300  CLIP OFF
```

Figure D11 (cont) TPL Program: TURB3

```
1310  CSIZE 3.0,.4
1320  LORG 6
1330  FOR I=Xa TO Xb STEP X_range/Dy
1340  MOVE I,Ya-.01*Y_range
1350  LABEL USING "#,MD.DD";I
1360  NEXT I
1370  LORG 8
1380  FOR I=Ya TO Yb STEP Y_range/Dy
1390  MOVE Xa-.01*X_range,I
1400  LABEL USING "#,MD.DD";I
1410  NEXT I
1420  K1=0
1430  !----- Trailing Edge -----
1440  T2=-A1
1450  T3=180-A1-E1
1460  T4=(T3-T2)/20
1470  FOR T1=T2 TO T3 STEP T4
1480  X=-R8*SIN(T1)
1490  Y=R8*(1-COS(T1))
1500  PLOT X,Y
1510  NEXT T1
1520  !----- Pressure Side -----
1530  X1=-(R1+R8)*SIN(A1+E1)
1540  Y1=R8+(R1+R8)*COS(A1+E1)
1550  T2=-90+A1+E1
1560  T3=B1
1570  T4=(T3-T2)/20
1580  FOR T1=T2 TO T3 STEP T4
1590  X=X1+R1*COS(T1)
1600  Y=Y1+R1*SIN(T1)
1610  PLOT X,Y
1620  NEXT T1
1630  !----- Tip Radius -----
1640  X4=X1+(R1+R9)*COS(B1)
1650  Y4=Y1+(R1+R9)*SIN(B1)
1660  T2=-B1
1670  T3=180-G1
1680  T4=(T3-T2)/20
1690  FOR T1=T2 TO T3 STEP T4
1700  X=X4-R9*COS(T1)
1710  Y=Y4+R9*SIN(T1)
1720  PLOT X,Y
1730  NEXT T1
1740  X6=X
1750  Y6=Y
1760  PENUP
1770  !----- Suction Side -----
1780  PEN 4
1790  X2=-(R2-R8)*SIN(A1)
1800  Y2=R8+(R2-R8)*COS(A1)
1810  T2=0
1820  T3=E1
1830  T4=(T3-T2)/20
1840  FOR T1=T2 TO T3 STEP T4
1850  X=X2+R2*SIN(T1+A1)
1860  Y=Y2-R2*COS(T1+A1)
1870  PLOT X,Y
1880  NEXT T1
1890  !----- Suction Side Radius R3 -----
1900  X5=X2+R2*SIN(E1+A1)
1910  Y5=Y2-R2*COS(E1+A1)
1920  X3=X5-R3*SIN(E1+A1)
1930  Y3=Y5+R3*COS(E1+A1)
1940  T2=-(90-A1-E1)
1950  T3=G1
```

Figure D11 (cont) TPL Program: TURB3

```
1960  T4=(T3-T2)/20
1970  FOR T1=T2 TO T3 STEP T4
1980  X=X3+R3*COS(T1)
1990  Y=Y3+R3*SIN(T1)
2000  IF Y<Y6 THEN 2060
2010  PENUP
2020  PRINT
2030  PRINT "These values give discontinuous slope on suction side"
2040  PRINT
2050  GOTO 370
2060  PLOT X,Y
2070  NEXT T1
2080  PLOT X6,Y6
2090  PENUP
2100  IF K1=1 THEN 2140
2110  K1=1
2120  WINDOW Xa-S0,Xb-S0,Ya,Yb
2130  GOTO 1430
2140  !----- Area Progression Thru Passage -----
2150  WINDOW Xa,Xb,Ya,Yb
2160  X7=λ1+S0
2170  Y7=Y1
2180  T2=-(90-A1-E1)
2190  T3=B1
2200  T4=(T3-T2)/20
2210  PEN 7
2220  LINE TYPE 5
2230  FOR T1=T2 TO T3 STEP T4*2
2240  X=X7+(R1-A0)*COS(T1)
2250  Y=Y7+(R1-A0)*SIN(T1)
2260  PLOT X,Y
2270  NEXT T1
2280  DUMP GRAPHICS
2290  DISP "F2 to continue "
2300  PAUSE
2310  CLEAR SCREEN
2320  INPUT "Calculate blade number? (1=Yes, 0=No): ",N0
2330  IF N0=0 THEN 2470
2340  PRINTER IS 711
2350  INPUT "Enter axial chord (in.): ",B
2360  INPUT "Enter mean radius (in.): ",R5
2370  S=b/B0
2380  Z1=2*PI*R5/S
2390  B=B0*S
2400  A=A0*S
2410  PRINT
2420  PRINT "Axial chord      = ";B;" in."
2430  PRINT "Blade space       = ";S;" in."
2440  PRINT "Throat width      = ";A;" in."
2450  PRINT "Number of blades = ";Z1
2460  PRINTER IS CRT
2470  DISP "F2 to exit, F3 for another run"
2480  PAUSE
2490  LOAD "DESIGN_MENU",10
2500  END
```

Figure D11 (cont) TPL Program: TURB3

```
10    ! Title:  TURB4
20    ! Author:  R. P. Shreeve (09/01/88)
30    ! Updated:  R. A. Wendland, LCDR, USN, (12/17/91)
40    ! Program:  Calculate blade height and lossess for given mean
50    !           line conditions for a single stage axial turbine
60    !
70    DIM O(3,17)
80    CLEAR SCREEN
90    PRINTER IS CRT
100   DUMP DEVICE IS 711
110   KEY LABELS OFF
120   !----- Design Input Data ---------------------------------------------
130   PRINT "Input data from program 4431T:"
140   INPUT "Enter Pt0 (psia): ",P0
150   INPUT "Enter Tt0 (deg R): ",T0
160   INPUT "Enter delta Tw (DELTW deg R): ",J0
170   INPUT "Enter w dot (lbs/sec): ",W0
180   !----- Gas Properties ------------------------------------------------
190   R4=53.393
200   G9=1.4
210   C9=.24
220   J9=778
230   G8=32.174
240   !----- Velocity Diagram Data -----------------------------------------
250   DEG
260   INPUT "Enter Alpha_e (deg): ",L3
270   INPUT "Enter delta alpha (deg): ",L4
280   INPUT "Enter M1cr: ",M1
290   INPUT "Enter Beta_e (deg): ",B3
300   INPUT "Enter delta beta (deg): ",B4
310   INPUT "Enter M'cr1: ",M3
320   INPUT "Enter M'cr2: ",M4
330   !----- Stator Blade Data ---------------------------------------------
340   CLEAR SCREEN
350   PRINT "Input STATOR blade data from TURB3:"
360   INPUT "Enter number of blades (z): ",Z1
370   INPUT "Enter throat width (a): ",A
380   INPUT "Enter axial chord (b): ",B
390   INPUT "Enter blade spacing (S): ",S
400   INPUT "Enter blade chord (c): ",C
410   INPUT "Enter trailing edge thickness (ts): ",T7
420   INPUT "Enter maximum blade thickness (tmax): ",T8
430   INPUT "Enter tip clearance (delta t): ",T9
440   !----- Analysis of the Stator ----------------------------------------
450   K=1
460   CLEAR SCREEN
470   INPUT "Enter STATOR profile loss (fig 15 (ref: 1174VA1)): ",S1
480   S9=2*S1
490   N0=1
500   X8=M1^2*(G9-1)/(G9+1)
510   X9=X8/(1-S9)
520   R0=P0*144/(R4*T0)
530   V0=SQR(2*C9*T0*G8*J9)
540   P5=(1-X9)^(G9/(G9-1))
550   T5=1-X8
560   T1=T5*T0
570   P1=P5*P0
580   R1=R0*P5/T5
590   !--- Re into stator
600   X6=X8*(COS(L3)/COS(L3-L4))^2
610   T=1-X6
620   P=T^(G9/(G9-1))
630   R=R0*P/T
640   J1=T*T0
650   M8=1.153E-5*.06333*SQR(J1)/(198.72/J1+1)
```

Figure D12 TPL Program: TURB4

```
660   R9=SQR(X6)*V0*R*C/(M8*12)
670   Q8=R0*V0*SQR(X9)*(1-X9)^(1/(G9-1))
680   D8=L4
690   D7=L3
700   M=SQR((2/(G9-1))*X8/(1-X8))
710   PRINT
720   PRINT
730   PRINT "STATOR Blade Analysis:"
740   PRINT "----------------------"
750   PRINT
760   GOSUB Blading_loss
770   IF ABS(S9-S6)<.002 THEN 840
780   PRINT "Above results are for interation ";N0
790   S9=S6
800   N0=N0+1
810   GOTO 510
820   PRINT
830   PRINT
840   PRINT "   Blade height = ";H9;" in."
850   PRINT "   Exit press.  = ";P1;" psia"
860   PRINT "   Exit temp.   = ";T1;" deg R"
870   PRINT
880   PRINT
890   PRINT
900   O(K,16)=P1
910   O(K,17)=T1
920   DISP "F2 to continue"
930   PAUSE
940   !----- Rotor Blade Data ----------------------------------------
950   K=2
960   PRINT "Input ROTOR blade data from TURB3:"
970   INPUT "Enter number of blades (z): ",Z1
980   INPUT "Enter throat width (a): ",A
990   INPUT "Enter axial chord (b): ",B
1000  INPUT "Enter blade spacing (S): ",S
1010  INPUT "Enter blade chord (c): ",C
1020  INPUT "Enter trailing edge thickness (ts): ",T7
1030  INPUT "Enter maximum blade thickness (tmax): ",T8
1040  INPUT "Enter tip clearance (delta t): ",T9
1050  INPUT "Enter ROTOR profile loss (fig 15, ref: 1174VA1): ",S1
1060  S9=3*S1
1070  N0=1
1080  G7=(G9-1)/(G9+1)
1090  Y8=M4^2*G7
1100  Y9=Y8/(1-S9)
1110  T4=(1-G7*M1^2)/(1-G7*M3^2)
1120  P4=(T4/T5)^(G9/(G9-1))
1130  R3=P4*P5*R0/T4
1140  W3=V0*SQR(T4)
1150  P6=(1-Y9)^(G9/(G9-1))
1160  T6=1-Y8
1170  P2=P6*P4*P1
1180  T2=T6*T4*T0
1190  !----- Re into ROTOR
1200  M8=1.153E-5*.06333*SQR(T1)/(198.72/T1+1)
1210  Y7=M3^2*G7
1220  R9=SQR(Y7)*W3*R1*C/(M8*12)
1230  Q8=R3*W3*SQR(Y9)*(1-Y9)^(1/(G9-1))
1240  D8=B4
1250  D7=B3
1260  M=SQR((T4/T5-1)*2/(G9-1))
1270  PRINT
1280  PRINT
1290  PRINT "ROTOR Blade Analysis:"
1300  PRINT "----------------------"
```

Figure D12 (cont) TPL Program: TURB4

```
1310  PRINT
1320  GOSUB Blading_loss
1330  IF ABS(S9-S6)<.002 THEN 1380
1340  PRINT "Above results are iteration ";N0
1350  S9=S6
1360  N0=N0+1
1370  GOTO 1100
1380  PRINT
1390  PRINT
1400  PRINT "    Blade height = ";H9;" in."
1410  PRINT "    Exit press.  = ";P2;" psia"
1420  PRINT "    Exit temp.   = ";T2;" deg R"
1430  PRINT
1440  PRINT
1450  PRINT
1460  O(K,16)=P2
1470  O(K,17)=T2
1480  DISP "F2 to continue"
1490  PAUSE
1500  !----- Calculate Stage Performance -----------------------------------
1510  K=3
1520  P8=P0/P2
1530  P9=P2*((T0-J0)/T2)^(G9/(G9-1))
1540  P9=P0/P9
1550  E8=T0*(1-(1/P8)^((G9-1)/G9))
1560  E8=J0/E8
1570  E9=T0*(1-(1/P9)^((G9-1)/G9))
1580  E9=J0/E9
1590  PRINT
1600  PRINT "Stage Performance:"
1610  PRINT "------------------"
1620  PRINT
1630  PRINT "   Pressure Ratio (T-S) = ";P8;"  Efficiency (T-S) = ";E8
1640  PRINT "   Pressure Ratio (T-T) = ";P9;"  Efficiency (T-T) = ";E9
1650  PRINT
1660  PRINT
1670  O(K,1)=P8
1680  O(K,2)=E8
1690  O(K,3)=P9
1700  O(K,4)=E9
1710  !----- Print Results -------------------------------------------------
1720  PRINT "Data to be printed on Printer."
1730  PRINT "-- Ensure paper is correctly set, press F2 to continue."
1740  PAUSE
1750  PRINTER IS 711
1760  PRINT
1770  PRINT
1780  PRINT TAB(27);"Single Stage Axial Turbine Design"
1790  PRINT TAB(27);"--------------------------------"
1800  PRINT
1810  PRINT
1820  PRINT TAB(41);"Stator";TAB(61);"Rotor"
1830  PRINT TAB(41);"------";TAB(61);"-----"
1840  PRINT
1850  PRINT USING "15X,K,11X,5D,15X,5D";"No. of Blades";O(1,8);O(2,8)
1860  PRINT USING "15X,K";"-------------"
1870  PRINT
1880  PRINT
1890  PRINT TAB(16);"Blade geometry (inches)"
1900  PRINT TAB(16);"--------------"
1910  PRINT
1920 Format1:  IMAGE 15X,K,5X,D.3D,14X,D.3D
1930  PRINT USING Format1;"Blade Height        ",O(1,7),O(2,7)
1940  PRINT USING Format1;"Exit Width          ",O(1,9),O(2,9)
1950  PRINT USING Format1;"Axial Chord         ",O(1,10),O(2,10)
```

Figure D12 (cont) TPL Program: TURB4

```
1960  PRINT USING Format1;"Spacing              ",O(1,11),O(2,11)
1970  PRINT USING Format1;"Chord                ",O(1,12),O(2,12)
1980  PRINT USING Format1;"T.E. Thickness       ",O(1,13),O(2,13)
1990  PRINT USING Format1;"Max. Thickness       ",O(1,14),O(2,14)
2000  PRINT USING Format1;"Tip Clearance        ",O(1,15),O(2,15)
2010  PRINT
2020  PRINT
2030 Format2: IMAGE 15X,K,5X,D.4D,13X,D.4D
2040  PRINT TAB(16);"Blade Row Performance"
2050  PRINT TAB(16);"--------------------"
2060  PRINT USING Format2;"Ref. Profile Loss    ",O(1,1),O(2,1)
2070  PRINT USING Format2;"Tot. Profile Loss    ",O(1,2),O(2,2)
2080  PRINT USING Format2;"Sec. Flow Loss       ",O(1,3),O(2,3)
2090  PRINT USING Format2;"Mixing Loss          ",O(1,4),O(2,4)
2100  PRINT USING Format2;"Tip Clearance Loss   ",O(1,5),O(2,5)
2110  PRINT USING Format2;"Total Loss Coeff.    ",O(1,6),O(2,6)
2120  PRINT
2130 Format3: IMAGE 15X,K,5X,3D.2D,14X,3D.2D
2140  PRINT USING Format3;"Exit Pressure (psia)",O(1,16),O(2,16)
2150  PRINT USING Format3;"Exit Temp.    (deg R)",O(1,17),O(2,17)
2160  PRINT
2170  PRINT
2180  PRINT TAB(16);"Stage Performance";TAB(39);"Total-Total";TAB(59);"Tot-Static"
2190  PRINT TAB(16);"-----------------";TAB(39);"-----------";TAB(59);"----------"
2200  PRINT
2210  PRINT USING Format1;"Pressure Ratio       ";O(3,3),O(3,1)
2220  PRINT USING Format2;"Efficiency           ";O(3,4),O(3,2)
2230  PRINTER IS CRT
2240  GOTO 2740
2250  !===== Blading loss subroutine ==========================================
2260 Blading_loss: !
2270  !----- Total Profile Loss -----
2280  PRINT "Re = ";R9
2290  INPUT "Enter K(Re)  (fig 23, ref: GA1074VA2): ",K1
2300  PRINT "Mach# = ";M
2310  INPUT "Enter K(M)  (fig 18, ref: 1174VA1): ",K3
2320  K2=1+2*(T8/C-.2)
2330  S2=S1*K1*K2*K3
2340  W9=W0*144/(Z1*Q8*A^2)
2350  H9=A*(W9*(1+.77*S2)+.025*S2*D8)
2360  !----- Secondary Flow Loss -----
2370  H5=(S-T7)*COS(D7)/H9
2380  S3=S2*.0323*D8*H5*(1+.962*S2)/(1-.03*D8*H5*S2)
2390  !----- Mixing Loss -----
2400  S4=(T7/S+.962*S2)/(1+.962*S2)
2410  S4=(COS(D7)*S4/(1-S4))^2
2420  S4=S4/(1+S4)
2430  !----- Tip Clearance Loss -----
2440  D6=D7-D8
2450  D5=ATN((TAN(D6)+TAN(D7))/2)
2460  S5=COS(D7)^2*(ABS(TAN(D6)-TAN(D7))^1.5/COS(D5))
2470  S5=2.26*S5*T9/(H9*SQR(B/S))
2480  S5=S5/(1+S5)
2490  !----- Total Losses -----
2500  S6=S2+S3+S4+S5
2510  PRINT "Ref. Profile Loss    = ";S1
2520  PRINT "Tot. Profile Loss    = ";S2
2530  PRINT "Sec. Flow Loss       = ";S3
2540  PRINT "Mixing Loss          = ";S4
2550  PRINT "Tip Clearance Loss   = ";S5
2560  PRINT
2570  PRINT "      Total Loss = ";S6
2580  O(K,1)=S1
2590  O(K,2)=S2
2600  O(K,3)=S3
```

Figure D12 (cont) TPL Program: TURB4

```
2610  O(K,4)=S4
2620  O(K,5)=S5
2630  O(K,6)=S6
2640  O(K,7)=H9
2650  O(K,8)=Z1
2660  O(K,9)=A
2670  O(K,10)=B
2680  O(K,11)=S
2690  O(K,12)=C
2700  O(K,13)=T7
2710  O(K,14)=T8
2720  O(K,15)=T9
2730  RETURN
2740  DISP "F2 to exit, F3 for another run"
2750  PAUSE
2760  PRINTER IS CRT
2770  LOAD "DESIGN_MENU",10
2780  END
```

Figure D12 (cont) TPL Program: TURB4

```
10    !Program: DESIGN_MENU
20    !Description: Program to provide Function Key menu selection of programs.
30    !
40    CLEAR SCREEN
50    PRINTER IS CRT
60    KEY LABELS ON
70    ON KEY 1 LABEL "R_4431T" GOTO R_4431t
80    ON KEY 2 LABEL "A_4431T" GOTO A_4431t
90    ON KEY 3 LABEL "TURB3" GOTO Turb3
100   ON KEY 4 LABEL "TURB4" GOTO Turb4
110   ON KEY 5 LABEL " " GOTO Hold
120   ON KEY 6 LABEL " " GOTO Hold
130   ON KEY 7 LABEL "MAIN     MENU" GOTO Main
140   ON KEY 8 LABEL "EXIT     MENU" GOTO Exit
150   !
160   PRINT "Turbo-propulsion Design Programs"
170   PRINT
180   PRINT "Item:                              Select Function Key"
190   PRINT
200   PRINT "  R_4431T: Performanc Calculations          F1"
210   PRINT "           (Iterate on R*)"
220   PRINT "  A_4431T: Performanc Calculations          F2"
230   PRINT "           (Iterate on Alpha1)"
240   PRINT "  TURB3: Draw Blade Profiles                F3"
250   PRINT "  TURB4: Blade Height and Loses             F4"
260   PRINT
270   PRINT "  Main Menu                                 F7"
280   PRINT "  Exit Menu                                 F8"
290   !
300 Hold:  !
310   GOTO Hold
320 R_4431t:  !
330   LOAD "R_4431T",10
340 A_4431t:  !
350   LOAD "A_4431T",10
360 Turb3: !
370   LOAD "TURB3",10
380 Turb4: !
390   LOAD "TURB4",10
400 Main:  !
410   MASS STORAGE IS "/WORKSTATIONS"
420   LOAD "AUTOST",10
430 Exit:CLEAR SCREEN
440   END
```

Figure D13 TPL Program: DESIGN_MENU

```
1     !Program:  SCAN
10    !Descrip:  Reads voltages from designated scanivalve through scanner #1
20    !          and prints to CRT port # and voltages in pressure (in. Hg).
30    !Devices:  HP3495A(701) #1, HP3456A(722), HG-78(707), Scanivalve(1-5)
40    !Modify:   Using Molye routines.
50    !Notes:    Devices HP3495A and HG-78 use specific communication formats
60    !          in the OUTPUT statements by utilizing the image form USING.
70    !
80    !             HP3495A: OUTPUT 701 USING "DDD";V+9
90    !             HG-78:   OUTPUT 707 USING "#,K";V
100   !
110   DIM Press(1:50)                          !Assign array in Memory.
120   !Designate Scanivalve:
130   INPUT "Input Scanivalve, First port, Last port:",V,First,Last
140   !
150   OUTPUT 722;"6STG10STIT3"                 !Set-up HP3456A DVM.
160   CLEAR 701                                !Reset HP3495A Scanner.
170   !
180   FOR Port_reqd=First TO Last              !Routine to read pressure values.
190     GOSUB Read                             !Subroutine to read Scanivalve
200     WAIT .5                                !port number from HG-78.
210     OUTPUT 701 USING "DDD";V+9             !Set Scanner connect DVM to HG-78
220     TRIGGER 722                            !Trigger DVM to read Scanivalve.
230     ENTER 722;Press(Port_reqd)             !DVM reads value, and writes to
240     CLEAR 701                              !HP9000 memory. Scanner is reset.
250   NEXT Port_reqd
260   CLEAR 722                                !Reset DVM.
270   !
280   PRINT "PORT #","PRESS (IN. H20)"         !Routine to print results to the
290   PRINT                                    !HP9000 CRT.
300   FOR I=First TO Last
310     PRINT I,Press(I)*100000
320   NEXT I
330   !
340   GOTO Finish
350   !**********END OF MAIN PROGRAM*********************************************
360   !
370   !**********SUBROUTINE READ AND POSITION SCANIVALVE ***********************
380 Read:  !
390   OUTPUT 707 USING "#,K";V                 !Routine to call HG-78, ask for the
400   P0=SPOLL(707)                            !current port assigned, read the port
410   L=BINAND(P0,15)                          !number, and convert it to decimal
420   T=SHIFT(P0,4)                            !format.
430   M=BINAND(T,7)
440   Port_read=10*M+L                         !Scanivalve port number in decimal
450   CLEAR 707                                !format.
460   IF Port_read=Port_reqd THEN Exit         !Exit subroutine if reqd port selected
470   OUTPUT 701 USING "DDD";V-1               !Advance S/V to next port
480   CLEAR 701                                !Reset Scanner
490   WAIT .1
500   GOTO Read                                !Loop to Read for another port reading.
510 Exit:RETURN
520   !**********END OF SUBROUTINES**********************************************
530 Finish:!
540   END
```

Figure D14 TPL Program: SCAN

```
10    SUB Plot
20    ! Subroutine to display plot screen, less the plots of any curves
30    ! for the specified variables in the COM /Plot_labels/ line.
40    COM /Plot_labels/ Xo,Xf,Yo,Yf,Dx,Dy,Title$,X_label$,Y_label$
50    CLEAR SCREEN
60    KEY LABELS OFF
70    GINIT                                !Initialize graph routine
80    X_range=Xf-Xo                        !Length of X-axis
90    Y_range=Yf-Yo                        !Length of Y-axis
100   LORG 6                               !Character ref pt:top center
110   MOVE 100*RATIO/2,100                 !Move cursor to screen loc for labels
120   CSIZE 3                              !Sizes labeling
130   LABEL Title$                         !Plot title
140   MOVE 100*RATIO/2,0                   !Move cursor to bottom center screen
150   LORG 4                               !Character ref pt:bottom center
160   LABEL X_label$                       !X-axis label
170   DEG                                  !Desig degrees for LDIR
180   LDIR 90                              !Sets Y-axis label on end
190   LORG 6
200   MOVE 0,50
210   LABEL Y_label$                       !Y-axis label
220   LDIR 0                               !Reset label to horizontal orientation.
230   LORG 2                               !Chr ref pt:left center
240   VIEWPORT 10,90*RATIO,10,90           !Sets graph screen size
250   FRAME                                !Box around VIEWPORT
260   WINDOW Xo,Xf,Yo,Yf                   !Set axis lengths in VIEWPORT
270   AXES X_range/Dx,Y_range/Dy,Xo,Yo        !Axes intersect at lower left
280   AXES X_range/Dx,Y_range/Dy,Xf,Yf        !Axes intersect at upper right
290   GRID X_range/Dx,Y_range/Dy,Xo,Yo,Dx,Dy,.001  !Create dot grid background
300   CLIP OFF                             !So labels can print outside VIEWPORT
310   CSIZE 3.0,.4                         !Axes label size
320   LORG 6                               !Number X-axis
330   FOR I=Xo TO Xf STEP X_range/Dx
340     MOVE I,Yo-.01*Y_range
350     LABEL USING "#,K";I
360   NEXT I
370   LORG 8                               !Number Y-axis
380   FOR I=Yo TO Yf STEP Y_range/Dy
390     MOVE Xo-.01*X_range,I
400     LABEL USING "#,K";I
410   NEXT I
420   CLIP ON
430   !
440   SUBEND
```

Figure D15 TPL Subprogram: Plot

```
10    !Date$=FNDate$(TIMEDATE)
20    DEF FNDate$(Seconds)
30      Julian=Seconds DIV 86400-1721119
40      Year=(4*Julian-1) DIV 146097
50      Julian=(4*Julian-1) MOD 146097
60      Day=Julian DIV 4
70      Julian=(4*Day+3) DIV 1461
80      Day=(4*Day+3) MOD 1461
90      Day=(Day+4) DIV 4
100     Month=(5*Day-3) DIV 153  ! Month
110     Day=(5*Day-3) MOD 153
120     Day=(Day+5) DIV 5        ! Day
130     Year=100*Year+Julian
140     IF Month<10 THEN
150       Month=Month+3
160     ELSE
170       Month=Month+3
180       Year=Year+1
190     END IF
200     Year$=VAL$(Year)
210     IF Month<10 THEN
220       Month$="0"&VAL$(Month)
230     ELSE
240       Month$=VAL$(Month)
250     END IF
260     IF Day<10 THEN
270       Day$="0"&VAL$(Day)
280     ELSE
290       Day$=VAL$(Day)
300     END IF
310     D$=Year$[3,4]&Month$&Day$
320     RETURN D$
330   FNEND
```

Figure D16 TPL Subprogram: FNDate$

```
10    !Program:FILE_XFER
20    !Description: transfers selected files to a selected directory, and
30    !             PURGES the same file if desired.
40    ON ERROR GOTO Error
50    CLEAR SCREEN
60    DIM Directory1$[35]
70    DIM Directory2$[35]
80    PRINT "This program transfers files from an existing specified INITIAL directory"
90    PRINT "to a NEW directory."
100   INPUT "Initial Directory? (ex. '/','/WORK/')",Directory1$
110   INPUT "New directory? (ex. '/','/WORK/')",Directory2$
120   CLEAR SCREEN
130   CAT Directory1$
140 Transfer:  !
150   INPUT "File for transfer (CAT to print directory, QUIT to Exit)",File$
160   IF File$="QUIT" THEN Finish
170   IF File$="CAT" THEN
180     CLEAR SCREEN
190     CAT Directory1$
200     GOTO Transfer
210   END IF
220 Copy:  !
230   COPY Directory1$&File$ TO Directory2$&File$
240   PRINT "File ";Directory1$&File$;" transferred to ";Directory2$&File$
250   INPUT "Purge this file from current directory? (Yes=1 No=0)",Del
260   IF Del=1 THEN
270     PURGE Directory1$&File$
280   END IF
290   GOTO Transfer
300 Error:  !
310   IF ERRN=54 THEN
320     PRINT "File: ";Directory2$&File$;" already exist."
330     INPUT "Do you want to purge? (0=No 1=Yes)",File_purge
340     IF File_purge=1 THEN
350       PURGE Directory2$&File$
360       PRINT "File: ";Directory2$&File$;" purged."
370       GOTO Copy
380     ELSE
390       PRINT "File: ";Directory2$&File$;" NOT purged."
400     END IF
410   END IF
420   GOTO Transfer
430 Finish:  !
440   CLEAR SCREEN
450   END
```

Figure D17 TPL Program: FILE_XFER

```
10    !Program: MAIN_MENU returns working window to the Main Menu
20    CONTROL CRT,5;4                !Reset screen color to Green.
30    MASS STORAGE IS "/WORKSTATIONS"
40    LOAD "AUTOST",10
50    END
```

Figure D18 TPL Program: MAIN_MENU

LIST OF REFERENCES

1. HP9000 Series 200/300 Computers, BASIC 5.0/5.1 Documentation Guide and Master Index, Hewlett Packard Company, 1987.

2. HP 9000 Series 300 Computers, Installation Reference, Hewlett Packard Company, 1988.

3. HP 9000 Series 200/300 Computers, Peripheral Installation Guide, Hewlett Packard Company, 1988.

4. HP 9000 Series 200/300 Computers, Installing and Maintaining the BASIC 5.0/5.1 System, Hewlett Packard Company, 1988.

5. HP 9000 Series 200/300 Computers, Using the BASIC 5.0/5.1 System, Hewlett Packard Company, 1987.

6. HP 9000 Series 200/300 Computers, BASIC 5.0/5.1 Programming Techniques, Vol. 1: General Topics, Hewlett Packard Company, 1987.

7. HP 9000 Series 200/300 Computers, BASIC 5.0/5.1 Programming Techniques, Vol. 2: Porting Information, Hewlett Packard Company, 1987.

8 HP 9000 Series 200/300 Computers, BASIC 5.0/5.1 Interfacing Techniques, Vol. 1: General Topics, Hewlett Packard Company, 1987.

9. HP 9000 Series 200/300 Computers, BASIC 5.0/5.1 Interfacing Techniques, Vol. 2: Specific Interfaces, Hewlett Packard Company, 1987.

10. HP 9000 Series 200/300 Computers, BASIC 5.0/5.1 Graphic Techniques, Hewlett Packard Company, 1987.

11. HP 9000 Series 200/300 Computers, BASIC Language Reference, Volume 1: A-N, Hewlett Packard Company, 1989.

12. HP 9000 Series 200/300 Computers, BASIC Language Reference, Volume 2: O-Z, Hewlett Packard Company, 1989.

13. Using HP BASIC for Instrument Control, A Self-Study Course (Volume 1), Hewlett Packard Company, 1988.

14. Using HP BASIC for Instrument Control, A Self-Study Course (Volume 2), Hewlett Packard Company, 1988.

15. Multiprogrammer Documentation and Software, 14753A Programming Reference Manual, Hewlett Packard Company 1988.

16. Instruction Manual, ZOC-14 with G-Sensor, Scanivalve Corp., 1989.

17. Instruction and Service Manual, CALSYS2000 PRESSURE CALIBRATOR and PSC2000 POWER AND SOLENOID CONTROL UNIT, Scanivalve Corp., 1990.

18. Installation and Assembly Level Service Manual for Model 6944A, Series 200 Multiprogrammer HP Model 6944A and Multiprogrammer Interface Card HP Model 98633A, Hewlett Packard Company, 1984.

19. Operating Manual, 500 Kilohertz Analog-to-Digital Converter Card HP Model 69759A, Hewlett Packard Company, 1984.

20. Operating Manual, High Speed Memory Card Model 69791A and Memory Expansion Card Model 69792A, Hewlett Packard Company, 1984.

21. Operating Manual, Counter/Totalizer Card HP Model 69775A, Hewlett Packard Company, 1984.

22. Operating Manual, Timer/Pacer Card Model 69736A, Hewlett Packard Company, 1984.

23. Lancaster, Don, TTL Cookbook, Howard W. Sams & Co. Inc., 1974.

24. McCarville, P. A., Hardware and Software Improvements to a Paced Data Acquisition System for Turbomachines, M.S.A.E Thesis, Naval Postgraduate School, Monterey, California, 1981.

25. Golden, W. L., Static Pressure Measurements of the Shock-Boundary Layer Interaction in Simulated Fan Passage, M.S.A.E. Thesis, Naval Post Graduate School, Monterey, California, 1992.

26. Operating Manual, Relay Output Card Model 69730A, Hewlett Packard Company, 1982.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Turbopropulsion Laboratory, Code AA/SF Naval Postgraduate School Monterey, CA 93943-5002	10
4.	Commander Naval Air Systems Command Washington, DC 20361 ATTN: AIR 536 T	1
5.	Commanding Officer Naval Air Propulsion Center Trenton, NJ 08628 ATTN: Steve Clauser	1
6.	Scanivalve Corporation P. O. Box 609005 San Diego, CA 92160	1
7.	Hewlett Packard Company 351 East Evelyn Avenue Mountain View, CA 94039-7051 ATTN: Ashley Weerpass	1
8.	Richard A. Wendland 880 Mallard Circle Arnold, MD 21012	3